Semi-Cooperative Learning in Smart Grid Agents

Prashant P. Reddy

December 2013
CMU-ML-13-114

MACHINE LEARNING
DEPARTMENT

Carnegie Mellon

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **DEC 2013**	2. REPORT TYPE	3. DATES COVERED **00-00-2013 to 00-00-2013**
4. TITLE AND SUBTITLE **Semi-Cooperative Learning in Smart Grid Agents**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Carnegie Mellon University,School of Computer Science,Machine Learning Department,Pittsurgh,PA,15213**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR’S ACRONYM(S) 11. SPONSOR/MONITOR’S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES		

14. ABSTRACT

Striving to reduce the environmental impact of our growing energy demand creates tough new challenges in how we generate and use electricity. We need to develop Smart Grid systems in which distributed sustainable energy resources are fully integrated and energy consumption is efficient. Customers, i.e., consumers and distributed producers, require agent technology that automates much of their decision-making to become active participants in the Smart Grid. This thesis develops models and learning algorithms for such autonomous agents in an environment where customers operate in modern retail power markets and thus have a choice of intermediary brokers with whom they can contract to buy or sell power. In this setting, customers face a learning and multiscale decision-making problem ? they must manage contracts with one or more brokers and simultaneously, on a finer timescale, manage their consumption or production levels under existing contracts. On a contextual scale, they can optimize their isolated selfinterest or consider their shared goals with other agents. We advance the idea that a Learning Utility Management Agent (LUMA), or a network of such agents, deployed on behalf of a Smart Grid customer can autonomously address that customer?s multiscale decision-making responsibilities. We study several relationships between a given LUMA and other agents in the environment. These relationships are semi-cooperative and the degree of expected cooperation can change dynamically with the evolving state of the world. We exploit the multiagent structure of the problem to control the degree of partial observability. Since a large portion of relevant hidden information is visible to the other agents in the environment, we develop methods for Negotiated Learning, whereby a LUMA can offer incentives to the other agents to obtain information that sufficiently reduces its own uncertainty while trading off the cost of offering those incentives. The thesis first introduces pricing algorithms for autonomous broker agents, time series forecasting models for long range simulation, and capacity optimization algorithms for multi-dwelling customers. We then introduce Negotiable Entity Selection Processes (NESP) as a formal representation where partial observability is negotiable amongst certain classes of agents. We then develop our ATTRACTIONBOUNDED- LEARNING algorithm, which leverages the variability of hidden information for efficient multiagent learning. We apply the algorithm to address the variable-rate tariff selection and capacity aggregate management problems faced by Smart Grid customers. We evaluate the work on real data using Power TAC, an agent-based Smart Grid simulation platform and substantiate the value of autonomous Learning Utility Management Agents in the Smart Grid.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**219**	

Semi-Cooperative Learning in Smart Grid Agents

Prashant P. Reddy

December 2013
CMU-ML-13-114

Machine Learning Department
School of Computer Science
Carnegie Mellon University, Pittsburgh

THESIS COMMITTEE
Manuela M. Veloso, Chair
Tom M. Mitchell
Stephen F. Smith
Yoky Matsuoka (Nest Labs)

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

This research was partially sponsored by the Office of Naval Research under MURI grant number N000140911031; by the Portuguese Science and Technology Foundation; and by Intelligent Automation, Inc. under grant number 654-1, prime sponsor DARPA under grant number FA8650-08-C-7812. The views and conclusions in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.

Keywords: Semi-Cooperative Learning, Negotiated Learning, Multiagent Learning, Online Learning, Reinforcement Learning, Sequential Decision-Making, Multiscale Decision-Making, Behavioral Game Theory, Distributed Agents, Machine Learning, Artificial Intelligence, Smart Grid Agents

Abstract

Striving to reduce the environmental impact of our growing energy demand creates tough new challenges in how we generate and use electricity. We need to develop Smart Grid systems in which distributed sustainable energy resources are fully integrated and energy consumption is efficient. Customers, *i.e.*, consumers and distributed producers, require agent technology that automates much of their decision-making to become active participants in the Smart Grid. This thesis develops models and learning algorithms for such autonomous agents in an environment where customers operate in modern retail power markets and thus have a choice of intermediary brokers with whom they can contract to buy or sell power.

In this setting, customers face a learning and *multiscale decision-making problem* – they must manage contracts with one or more brokers and simultaneously, on a finer timescale, manage their consumption or production levels under existing contracts. On a contextual scale, they can optimize their isolated self-interest or consider their shared goals with other agents. We advance the idea that a *Learning Utility Management Agent* (LUMA), or a network of such agents, deployed on behalf of a Smart Grid customer can autonomously address that customer's multiscale decision-making responsibilities.

We study several relationships between a given LUMA and other agents in the environment. These relationships are *semi-cooperative* and the degree of expected cooperation can change dynamically with the evolving state of the world. We exploit the multiagent structure of the problem to control the degree of partial observability. Since a large portion of relevant hidden information is visible to the other agents in the environment, we develop methods for *Negotiated Learning*, whereby a LUMA can offer incentives to the other agents to obtain information that sufficiently reduces its own uncertainty while trading off the cost of offering those incentives.

The thesis first introduces pricing algorithms for autonomous broker agents, time series forecasting models for long range simulation, and capacity optimization algorithms for multi-dwelling customers. We then introduce *Negotiable Entity Selection Processes* (NESP) as a formal representation where partial observability is negotiable amongst certain classes of agents. We then develop our ATTRACTION-BOUNDED-LEARNING algorithm, which leverages the variability of hidden information for efficient multiagent learning. We apply the algorithm to address the *variable-rate tariff selection* and *capacity aggregate management* problems faced by Smart Grid customers. We evaluate the work on real data using Power TAC, an agent-based Smart Grid simulation platform and substantiate the value of autonomous Learning Utility Management Agents in the Smart Grid.

To my dear wife Heather,

our Akiva, Kellan & Carina,

and

my parents Jayasree & Jagathpal,

for your boundless love and untold sacrifice.

Acknowledgements

I owe thanks to all who made this journey possible and to all who made it worthwhile and enjoyable. Foremost amongst them is my advisor. Manuela, thank you for believing in me even though I knew nothing when I started down this road! I am extremely grateful for your guidance and support, and also for sharing your passion for AI, for robots, and for the human world around us. I am also indebted to ONR, PSTF and DARPA for funding this research and to IAI and the CMU|Portugal program for collaborating with us.

I have been fortunate to assemble a fantastic thesis committee. Tom, your singular vision for the discipline of machine learning over the next 100 years attracted me to CMU. I am honored to have you endorse this thesis as a contribution to that discipline. Steve, your curiosity and advice have made this thesis stronger in ways that I may not have fully conveyed to you, thank you. Yoky, having some understanding of the hectic pace of startup life, I cannot express how much I appreciate you taking the time to participate on this committee, for inviting me into your world of ideas, and for encouraging me every step of the way!

Guy Chiarello, Jeff Birnbaum, Steve Lieblich, and Ben Fried – I am tremendously thankful for your support of my adventure. I hope that our paths cross again in the near future.

During my first few days at CMU, the following people along with Tom and Manuela demonstrated that they are warm and welcoming colleagues first and brilliant scientists second: Steve Fienberg, Manuel Blum, and Reid Simmons – thank you.

Larry Wasserman's email address proclaims that he is *cool*—I have not met many people that embody the spirit of coolness better than him. Interacting with Geoff Gordon has convinced me that it is possible for one person to know something about everything that's worth knowing. I am thankful that I was able to tap into that knowledge in bits and pieces through the speaking skills committee. Inês Azevedo, thank you for serving on my DAP committee and for educating me on the purpose and challenges of energy sustainability.

I fully expected to form many research relationships through this PhD program, but I did not expect that some of the deepest collaborations and friendships would be formed across continents. Wolf Ketter and John Collins, thank you for sharing your insight, and for daring to create a virtual world where novel science can help create a more sustainable real world.

If I have one regret about my research, it is that I did not spend more time building robots. However, I have indulged vicariously through Manuela's CORAL group: Joydeep Biswas, Susana Brandão, Brian Coltin, Tom Kollar, Som Liemhetcharat, Francisco Melo, Çetin Meriçli, Stephanie Rosenthal, Mehdi Samadi, Junyun Tay, Felipe Trevizan, Richard Wang, and Stefan Zickler—thank you all for the experience.

The journey is only as enjoyable as your fellow travelers. Andrew Arnold, Andy Carlson, Polo Chau, Robert Fisher, Alona Fyshe, Min Xu, and Brian Ziebart—thank you for your friendship and for being terrific sources of support and motivation.

Diane Stidle, you have turned the machine learning department into a home away from home for many of us. I especially appreciate your ability and willingness to listen to ramblings about the minutiae of work and life as if each of them defined our existence—thank you!

Dave Mawhinney, thank you for everything you have done to inspire and enable me, and others like me, to pursue our dreams. Thank you for personifying all that is positive in a friend.

Mom and Dad, your sacrifices have given me opportunities that I am endlessly grateful for. Thank you also for providing a loving home, for instilling an appreciation of our planet, and for nurturing my interest in building things. Ravi Reddy, thanks for being the coolest uncle I have—you have been a role model ever since I can remember. Raj Venkat, thank you for always being there for me and the family. I cannot acknowledge all by name, but let that not diminish my gratitude for the support of my sisters, in-laws, and the rest of my family. Hy and Ellen Gillman, the ways that you have supported us over the past few years are too numerous to recount here. I deeply appreciate everything you have done to make this all possible.

Akiva, Kellan, and Carina, you are delightful in so many ways. I have come to realize the awesome power of machine learning through this PhD program, but watching you grow has only made me realize how much more awesome human learning is. You have been a source of profound joy and pride throughout this experience.

Heather, thank you for believing in me and in this journey. I love you and I hope you know how much I appreciate every little and big thing you have done to make this dream a reality.

Happy is the man who finds a true friend,
and far happier is he who finds that friend in his wife.
– Franz Schubert

Contents

List of Figures

List of Tables

List of Algorithms

List of Definitions

Chapter 1

Introduction

Striving to reduce the environmental impact of our growing energy demand creates tough new challenges in how we generate and use electricity. We need to develop new control systems that allow for efficient energy consumption and for distributed sustainable energy resources to be fully integrated into the power grid. Customers, *i.e.*, both consumers and distributed producers, require agent technology that automates the decision-making that is expected of them in such control systems. This thesis presents models and learning algorithms for such autonomous agents.

1.1 Energy Sustainability

Smart Grid refers to a loosely defined set of technologies aimed at modernizing the power grid using digital communications [Kannberg et al., 2003]. Prevailing power grid technology was mostly designed for one-way flow of electricity from large centralized power plants to distributed consumers such as households and industrial facilities. The Smart Grid aims to increase usage of distributed renewable energy resources, such as small wind farms or households with solar panels, by enabling them to efficiently sell power into the grid. It also aims to shift net demand for electricity to time periods when power is produced more cheaply. The corresponding increased complexity creates the need for new technical and economic control mechanisms.

1.1.1 Smart Grid Tariff Markets

One approach to addressing the challenge of increased participation from distributed producers is through the introduction of *brokers* who buy power from those producers and also sell power to consumers [Block et al., 2010]. Brokers interact with producers and consumers through a *tariff*

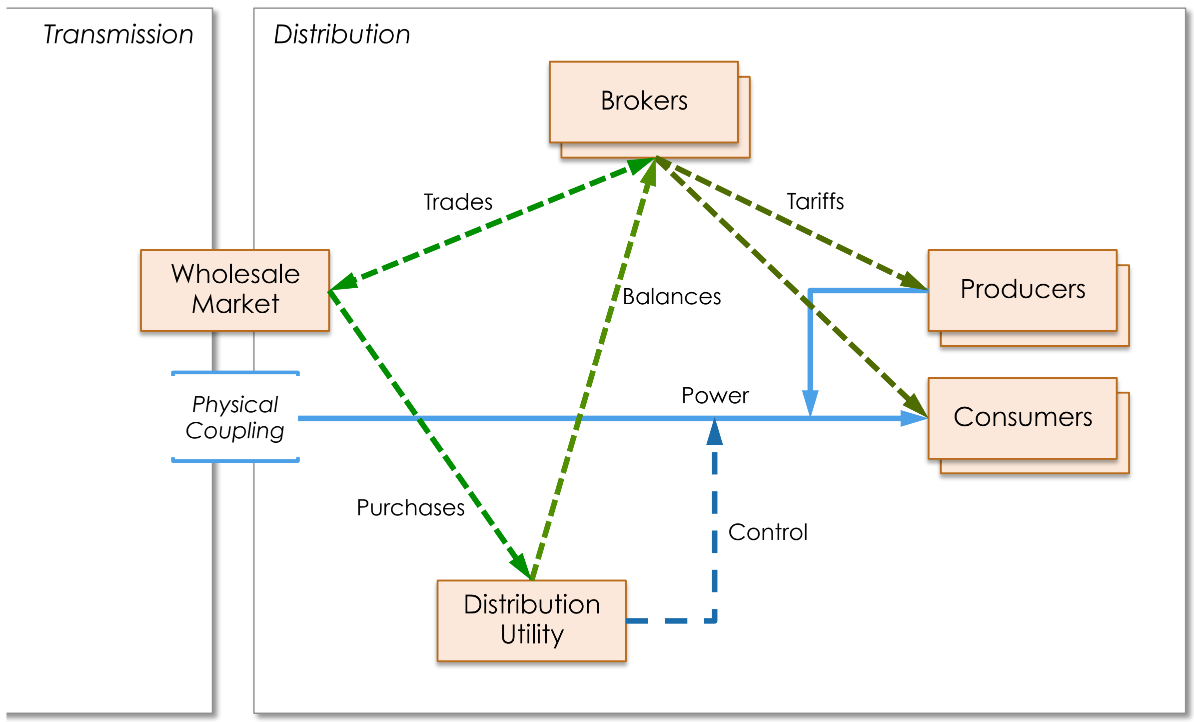


Figure 1.1: Architecture of a typical Smart Grid tariff market—consumers and producers buy/sell power by subscribing to tariffs offered by brokers.

market—a new retail market mechanism.[1] Figure 1.1 presents an overview of interactions in a Smart Grid tariff market. In this mechanism, each broker acquires a portfolio of producers and consumers by simultaneously publishing consumption and production tariffs.[2]

The design of fees and penalties in the tariff market incentivizes brokers to balance supply and demand within their portfolio by buying production and storage capacity from local producers instead of acquiring all supply from the transmission grid. This also gives them the ability to differentiate themselves in the market by offering prices distinct from those on the wholesale market while also helping local producers. As brokers compete in the tariff market, we expect them to tailor tariffs to appeal to specific segments of the customer population. In response, customers are likely to gradually migrate away from the default tariff offered by the monopoly distribution utility (DU), which would continue to operate the physical grid infrastructure.

[1]Retail power markets contrast with wholesale power markets or exchanges which were introduced about a decade ago to increase supply-side competition by separating large centralized power producers from monopoly distribution utilities.

[2]Tariffs are as-is contracts that customers can subscribe to without negotiation. In alternate models, brokers may also negotiate individual contracts containing customized terms with certain large customers.

1.1.2 *Problem*: Customer Decision-Making

The influx of new tariffs in the Smart Grid retail markets creates opportunities and challenges for customers. The availability of tariffs with distinct rate characteristics allows them to choose those that are best suited to their consumption or production characteristics. For example, time-of-use (TOU) tariffs vary rates depending on time-of-day, day-of-week and month-of-year whereas critical peak pricing (CPP) tariffs have higher rates at a few critical periods but offer lower rates otherwise. Increasingly, real-time pricing (RTP) tariffs, where rates fluctuate in close correlation with wholesale market prices, are gaining popularity because they more accurately convey the cost of power supply at any given time. Within these different tariff types, different brokers are likely to offer varied contract terms, such as signup bonuses, exit penalties, and tiered rates.

Variations in tariffs also introduce a significant challenge for customers. To evaluate the offered tariffs, customers must have a quantifiable understanding of their own consumption and production characteristics. Moreover, since many tariffs are explicitly designed to influence some modification of customer behavior away from their default behavior, customers must also understand how they plan to respond to these influences. For example, this may involve reducing consumption or shifting it to cheaper periods, or investing in battery storage to separate local production from when that power is sold to the grid. Therefore, customers must analyze their ability to best respond to the incentives offered by the tariffs to which they are subscribed. Furthermore, they must subscribe to one of the better suited tariffs given their anticipated behavior while simultaneously adjusting their behavior under the tariff to which they are currently subscribed, thus introducing a multi-timescale decision-making problem for customers.

Adding to the complexity, customers must also anticipate how other customers are likely to react to the tariffs in order to avoid the herding behavior that results from all of them responding similarly. Customers have **semi-cooperative** relationships with other customers since their goals are not necessarily aligned nor are they diametrically opposed. Thus, customers must choose the extent to which they collaborate with their *neighbors*. This introduces a multi-context decision-making problem in which each customer must balance individual goals with shared goals. This *contextual* dimension taken along with the *temporal* dimension from above leads to a multiscale decision-making problem for Smart Grid customers.

1.2 Thesis Approach

Various game-theoretic, learning-based and optimization techniques are applied in current research on active customer participation in Smart Grid control. The focus of such research has

been on strategies for large customers to directly participate in wholesale power markets and for smaller customers to form consumer cooperatives and virtual power plants (VPP). The broker-based tariff market setting that we consider generalizes those problems since a large customer can act as their own broker whereas cooperatives and VPPs can collaborate to create their own trusted brokers.

1.2.1 Autonomous Customer Agents

We advance the idea that a learning agent deployed on behalf of a Smart Grid customer can autonomously address that customer's multiscale decision-making responsibilities. The agent may be physically instantiated as a household device, a software component in a larger system, or a network of such devices and components. We refer to the possible manifestations uniformly as a *Learning Utility Management Agent* (LUMA). This thesis examines three sets of relationships from a LUMA's perspective (Figure 1.2):

1. To the associated customer – This relationship can be further separated into (i) the *customer delegate*, and (ii) one or more *capacity originators*. Each capacity originator is a consumption or production appliance (*e.g.*, air conditioning system, roof-top solar installation) or a customer sub-entity (*e.g.*, unit in an apartment building, industrial process in a factory).

2. To the brokers – While complex interactions are possible in scenarios where large customers negotiate custom contracts with brokers, we focus instead on subscription-based interactions that rely on published tariff contracts.

3. To other customers or their agents – We assume that there exists some mechanism by which a LUMA can discover and communicate with customers in its *neighborhood*, an abstract

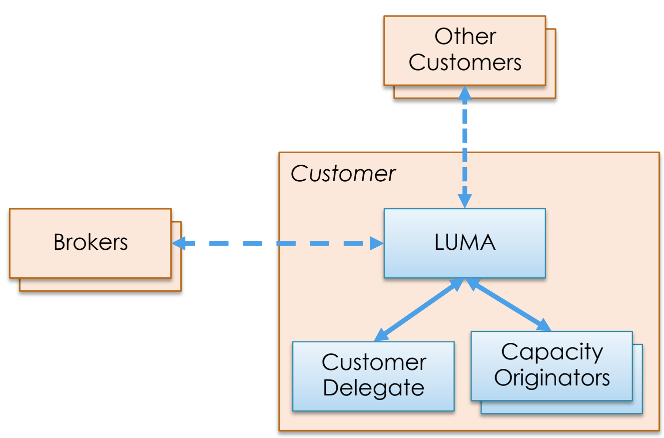


Figure 1.2: We study interactions between a Learning Utility Management Agent (LUMA) deployed on behalf of a Smart Grid customer and various other agents in the environment.

context that includes any customers whose behavior has a material impact on the goals of the LUMA's associated customer.

The purpose of a LUMA is to achieve the goals of the associated customer as conveyed by limited communication with the customer delegate. The other relationships are all *semi-cooperative* and the degree of cooperation can change dynamically with the evolving state of the world. We assume that the relationships with capacity originators are semi-cooperative because the LUMA represents the joint goals of the whole customer whereas each capacity originator may itself be an independently controlled self-interested agent. The relationships with brokers can be cooperative under specific situations but are largely adversarial in purely tariff-based interactions. Relationships with other customers can vary widely from cooperative to adversarial depending on the relative impact on tariff prices due to the actions of each customer.

Within the context of a LUMA defined as above and its associated environment, this thesis addresses the following question:

> *How can the multiscale decision-making tasks of a Smart Grid customer be addressed by an autonomous learning agent in a distributed agent environment?*

1.2.2 Negotiating with a Multiagent Oracle

Partially observable stochastic games (POSG) offer abstractions to model the various interactions in a LUMA's environment; however, corresponding game-theoretic and multiagent reinforcement learning algorithms pose severe computational challenges. To address such challenges, we can exploit the multiagent structure of the problem to control the degree of partial observability.

We assume that a significant portion of relevant hidden information is visible to the other agents in the environment. Then, all of the agents together can be viewed as an *oracle*, albeit an *incomplete* and *imperfect* one—incomplete in that the oracle does not necessarily include *all* hidden information and imperfect in that the information provided by the oracle may be subject to uncertainty and error. This insight enables us to develop **Negotiated Learning**, a technique whereby a LUMA can offer incentives to the other agents to obtain information that sufficiently reduces its own uncertainty while trading off the cost of offering those incentives. The incentives may be financial payments, non-monetary incentives such as mutual exchange of information, or disincentives associated with taking actions that somehow inconvenience any of the agents.

1.3 Thesis Overview

This thesis makes contributions not only to the field of computational energy sustainability but also to the discipline of machine learning (ML) and, more generally, artificial intelligence (AI). Specifically, we introduce models and algorithms for a class of multiagent semi-cooperative partially observable sequential decision-making problems. We apply them to develop intelligent strategies for Smart Grid customer agents. However, to fully understand the context for the problem setting, we also study various components of the Smart Grid tariff market as illustrated in Figure 1.1. Table 1.1 summarizes the contributions of this thesis to energy sustainability and Table 1.2 summarizes the ML and AI techniques used therein.

Learning Broker Agents

Our work on *broker strategies* is presented in **Chapter 2**. We develop a representation of the tariff market domain and the profit-maximizing goal of brokers as a Markov decision process (MDP). We focus on strategies that can be adopted by an autonomous agent representing a broker's goals. We first introduce adaptive strategies that leverage our MDP state features and outperform the simple fixed strategies prevalent in the real world. We then develop reinforcement learning strategies that outperform the adaptive strategies. We also study, using self-play, the economic impact of multiple brokers employing such learning strategies and find that all learning brokers outperform non-learning brokers. The ability to predict prices on wholesale electricity markets can be integrated into more comprehensive broker strategies, so we also analyze prices on the Ontario wholesale market using graphical and time series models.

Customer Model Simulation

Since Smart Grid tariff markets are yet to be implemented in much of the real world, and where they exist they are nascent, we rely on agent-based simulation to develop and validate the contributions of this thesis, modeling future markets and agent behaviors using real world data where

Table 1.1: Contributions to computational energy sustainability.

Component	Formulation	Model	Design	Algorithm	Analysis
Broker strategies	✓	✓		✓	✓
Wholesale markets		✓			✓
Simulation platform			✓		✓
Customer simulation		✓	✓		✓
Customer strategies	✓	✓		✓	✓

Table 1.2: Machine learning and AI techniques used in our work.

Component	**ML / AI techniques**
Broker strategies	– Reinforcement learning – Agent-based computational economics
Wholesale markets	– Time series analysis – Support vector machines
Simulation platform	– Distributed agent architecture – Mechanism design
Customer simulation	– Time series forecasting – Hierarchical Bayesian models + Gibbs sampling
Customer strategies	– Decision-theoretic stochastic optimization – Iterated strategic reasoning – *Negotiated Learning*

possible. In the development of such a simulation environment, we encounter the problem of time series simulation based on prior sample data, online *bootstrap* data, and subjective biases that must be introduced to simulate specific behaviors. We address this problem using a novel hierarchical Bayesian time series simulation method, which we describe in **Chapter 3** along with a brief overview of Power TAC, the encompassing simulation platform.

Adaptive Customer Agents

Simulation of the vastly heterogenous behaviors of Smart Grid customers of various sizes (*e.g.*, residential *vs.* commercial) and functions (*e.g.*, consumer *vs.* producer) is a key challenge in the Power TAC platform. We have developed a *factored customer model* framework to represent and simulate the production and consumption capacities of a diverse set of customer types. We use a generalized set of *factors*, many of them represented as probability distributions, to define the intrinsic behaviors of various customer types and their responses to stimuli from the simulation environment. This framework and some example instantiations are described in **Chapter 4**, which also delves into our first customer agent strategy, a decision-theoretic approach using stochastic optimization. We tackle the scenario of multi-dwelling consumers, such as apartment buildings and rural electricity cooperatives, where each dwelling maintains autonomy over its consumption behavior but is cooperative with the other dwellings to obtain lower costs and various shared benefits. We introduce a centralized *utility optimizer*, which models its interaction with each dwelling using the game-theoretic notion of *quantal response*. The resulting *approximate quantal response equilibrium* (ϵ-QRE) shows that the dwellings can autonomously maximize their own self interest and yet achieve cost savings and lower aggregate demand volatility.

Learning Customer Agents

Chapter 5 introduces *Negotiated Learning*—a novel approach that we use to develop learning Smart Grid customer agents. We first formulate the *variable rate tariff selection* problem, which characterizes the class of problems that can be addressed by Negotiated Learning. We then identify the dynamic multiagent structure of the problem, define a representation that captures the relevant structure, and present an algorithm that exploits such structure to solve the problem. We then formulate a second Smart Grid customer agent problem, *capacity aggregate management*, that also exhibits similar structure and can be addressed by Negotiated Learning. We conceptually explore the applicability of Negotiated Learning beyond the Smart Grid domain at the end of this chapter. **Chapter 6** presents experimental analysis of our Negotiated Learning methodology, including the ATTRACTION-BOUNDED-LEARNING algorithm, to validate its effectiveness in the development of learning Smart Grid customer agents.

This thesis draws upon research from a number of fields including computational energy sustainability, machine learning, multiagent systems, and behavioral game theory. Such related work is surveyed and contextualized in **Chapter 7**. Finally, **Chapter 8** concludes with a review of the contributions of this thesis and ideas for future work.

Note: Terminology and Notation

The Smart Grid domain relies upon extensive terminology that we summarize in Appendix B. Because of our focus on the intersection of computational agents, economics, and power systems, we are sometimes faced with difficult choices amongst terms used to identify similar concepts in the different disciplines. We have tried to explain and resolve these conflicts in Appendix B, the list of definitions, or within the chapters.

Moreover, the numerous models and algorithms that we have developed engender extensive notation—we provide a guide to the notational conventions and a list of symbols in Appendix A. The combination of Smart Grid terminology and the solution techniques also introduce a list of abbreviations that we collect in Section A.3.

Chapter 2

Broker Agents in Tariff Markets

This chapter provides some background on tariff markets and develops a Markov decision process representation of the domain from the perspective of a profit-maximizing goal broker. In Section 2.2, we focus on strategies that can be adopted by an autonomous agent representing a broker. We first introduce adaptive strategies that leverage our MDP state features and outperform the simple fixed strategies prevalent in the real world. We then develop learning strategies that outperform the adaptive strategies. In Section 2.2.3, we use self-play to study the economic impact of multiple brokers employing such learning strategies and find that all learning brokers outperform non-learning brokers. The ability to predict prices on *wholesale electricity markets* can be integrated into more comprehensive broker strategies, so we also analyze the movement of prices on a representative market and introduce a model for price prediction in Section 2.3.

2.1 Structure of Retail Power Grids

Current power grid architecture, illustrated in Figure 2.1, can be characterized as largely hierarchical. In the transmission grid, centralized high-voltage control centers manage the production schedules for large power plants based on demand forecasts. Such forecasts are typically estimated using historical demand observations, weather forecasts, long-term purchase contracts and trading in wholesale electricity markets. Distribution utilities (DU) purchase power, in the form of forward contracts, in the wholesale markets for delivery to their customers in the distribution grid. Because of their limited ability to store electricity, DUs must balance supply and demand very closely. Longer term imbalances are typically handled through additional purchases on the wholesale markets at short notice, which typically result in significant financial costs. Short term imbalances can significantly disrupt the flow of power on the grid, potentially leading to severe

outages, so they are handled by exerting discretionary control over a small set of production and consumption capacities that can be turned on/off or up/down on very short notice. Owners of such dedicated controllable capacities command significant price premiums for allowing the DU to control their capacities and therefore relying on such capacities tends to be even more expensive than short notice purchases on the wholesale markets [Skytte, 1999].

Demand-side management (DSM) capabilities on customer premises, such as water heaters, pool pumps and air-conditioners that can be shut off temporarily through remote control, offer an attractive alternative towards managing short term imbalances. However, consumers generally have very little awareness of these capabilities and are wary of remote controllability, so they must be incentivized to actively participate in the grid.

Effective use of intermittent energy sources such as wind and solar requires that consumers adapt to the cost and availability of renewable energy. A market structure that reflects the cost of power production will motivate many households and businesses to invest in some combination of demand management (*e.g.*, price-sensitive appliance controls), supply resources (*e.g.*, rooftop solar installations), and energy storage (*e.g.*, electric vehicle batteries). These newly-integrated capabilities will introduce noticeable demand elasticity.

Some proposals envision retail customers, or even their appliances, directly participating in the wholesale markets [Ramchurn et al., 2012]. However, wholesale markets are not designed to provide power for immediate delivery, nor are they designed to deal with large numbers of small-scale participants. Only a few large industrial customers possess the scale and financial wherewithal to participate directly in wholesale markets.

Retail *brokers* play the role of financial intermediaries, aggregating the demand (and supply) of large numbers of smaller customers, observing and forecasting their aggregate consumption and production patterns, and actively participating in the wholesale markets to minimize their risk-adjusted costs. Such a broker, acting on behalf of a large number of individual customers, can provide power at a lower average price, while making a profit, than the individuals could obtain on their own [Ketter et al., 2013]

Brokers interact with retail consumers and distributed producers in two ways:

1. *Tariffs*: Each customer has the option to *subscribe* to a *tariff* published by a broker.

   Definition 2.1: A **tariff** is a standardized agreement, as defined by its associated *tariff contract*, to buy or sell electricity to be delivered in the retail distribution grid.

   Definition 2.2: A **tariff contract** stipulates various terms and conditions, including fixed or variable rate specifications, contract periods, signup bonuses, early termination fees,

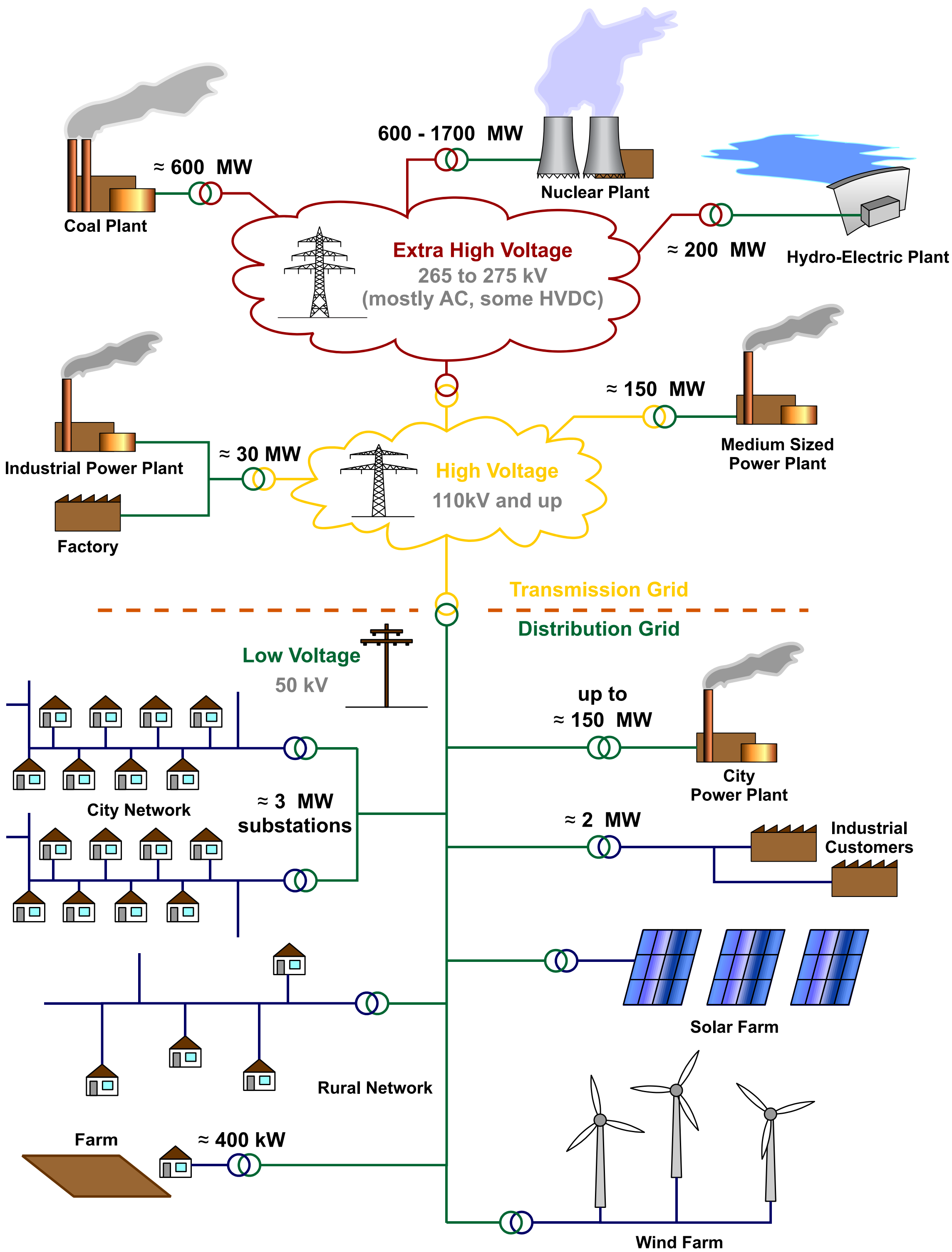


Figure 2.1: Overview of the physical structure of retail power grids. *Source: MDizon/CC-BY-3.0.*

periodic fees, and renewable energy content. We elaborate on the structure and ontology of tariff contracts in future chapters.

Definition 2.3: A **tariff subscription** accepts a tariff without modification of the associated tariff contract.

2. *Negotiated contracts*: Brokers negotiate customized contracts with larger customers who may have specific needs or demands that can be individually accommodated by the brokers.

We focus our research on tariff-based interactions. We refer to the resulting market mechanism as a *tariff market*, which we define in the next section.

2.2 Learning Broker Agent Strategies

As a first step in our research, we study the learning of pricing strategies for autonomous broker agents in tariff markets. We develop a broker agent that learns its strategy using reinforcement learning. We contribute methods for representing the tariff market domain and broker agent goal as a scalable Markov decision process (MDP) for Q-LEARNING. We also contribute a set of pricing tactics that form actions in the learned MDP policy.

2.2.1 *Problem*: Balancing in Tariff Markets

Definition 2.4: A **tariff market**, which operates over the distribution grid, is not an operating entity like the wholesale market but is instead defined by a set of market participants and rules. It consists of the following participants:

$$\langle \mathcal{C}, \mathcal{P}, \mathcal{B}, DU \rangle$$

where:

- $\mathcal{C} = \{C_j : j = 1..|\mathcal{C}|\}$ are the *Consumers* and $|\mathcal{C}| = \mathrm{O}(10^5)$;
- $\mathcal{P} = \{P_j : j = 1..|\mathcal{P}|\}$ are the *Producers* and $|\mathcal{P}| = \mathrm{O}(10^3)$;
- $\mathcal{B} = \{B_j : j = 1..|\mathcal{B}|\}$ are the *Broker Agents* and $|\mathcal{B}| = \mathrm{O}(10^1)$;
- DU is the *Distribution Utility*, a regulated regional monopoly that manages the physical infrastructure for the grid.

$\mathcal{C} \cup \mathcal{P}$ forms the combined set of potential customers from a broker agent's perspective.

Definition 2.5: A **tariff price** is a measure in $\mathbb{R}^+$ of the utility value of a tariff considering its rate specification and other contract terms, evaluated assuming uniform customer preferences over those terms.

The performance of a broker agent is evaluated over a finite time sequence, $\mathcal{T}$. At each time step $t \in \mathcal{T}$, each broker agent B_k publishes two tariffs—a *producer tariff* with price $p_{t,P}^{B_k}$ and a *consumer tariff* with price $p_{t,C}^{B_k}$. We assume in this chapter that these tariff prices are visible to all agents in the environment.

Each broker agent holds a *portfolio*, $\Psi_t = \Psi_{t,C} \cup \Psi_{t,P}$, of consumers and producers who have subscribed to one of its tariffs at the current time, t. Each consumer consumes a fixed amount of power, κ, per time step and each producer generates $\nu\kappa$ units of power per time step.

At each t, the *profit*, $r_t^{B_k}$ of a broker agent is the net proceeds from consumers, $\Psi_{t,C}$, minus the net payments to producers, $\Psi_{t,P}$, and the distribution utility:

$$r_t^{B_k} = \underbrace{p_{t,C}^{B_k}\,\kappa\,\Psi_{t,C}}_{\text{consumer payments}} - \underbrace{p_{t,P}^{B_k}\,\nu\,\kappa\,\Psi_{t,P}}_{\text{supplier payments}} - \underbrace{\phi_t\,|\kappa\,\Psi_{t,C} - \nu\,\kappa\,\Psi_{t,P}|}_{\text{balancing costs}} \tag{2.1}$$

Definition 2.6: The **balancing fee**, ϕ_t, specified by the distribution utility at each t, is used to penalize the supply-demand imbalance in a broker's portfolio at time t.

The term $|\kappa\Psi_{t,C} - \nu\kappa\Psi_{t,P}|$ represents the imbalance. The primary goal of a broker agent is to maximize its cumulative profit over all time steps, $\sum_{\mathcal{T}} r_t^{B_k}$.

2.2.1.1 Data-driven Simulation Model

We developed a simulation model that is driven by real-world hourly electricity prices from the Independent Electricity System Operator (IESO), a wholesale market in Ontario, Canada [IESO, 2011]. Each time step in simulation defines the smallest unit of time over which the tariff prices offered by a broker agent must be held constant. However, when considering the price to offer at each time step, a broker agent may use forecasted prices over a longer time horizon, H. For instance, the broker agent can take the moving average over weekly expected market prices and offer that as his producer tariff price for the next time step. Indeed we adopt this **Fixed** strategy in our model to simulate each broker agent. The consumer tariff price is then computed by adding a variable profit margin, μ. Figure 2.2 shows 4 producer tariff price sequences over 240 time steps; these are 4 of 50 distinct sequences derived from the real-world hourly data.

Each customer is represented by a *customer model*, which given an unordered set of tariffs returns a ranking according to its preferences. Customer models do not simply rank the tariffs by

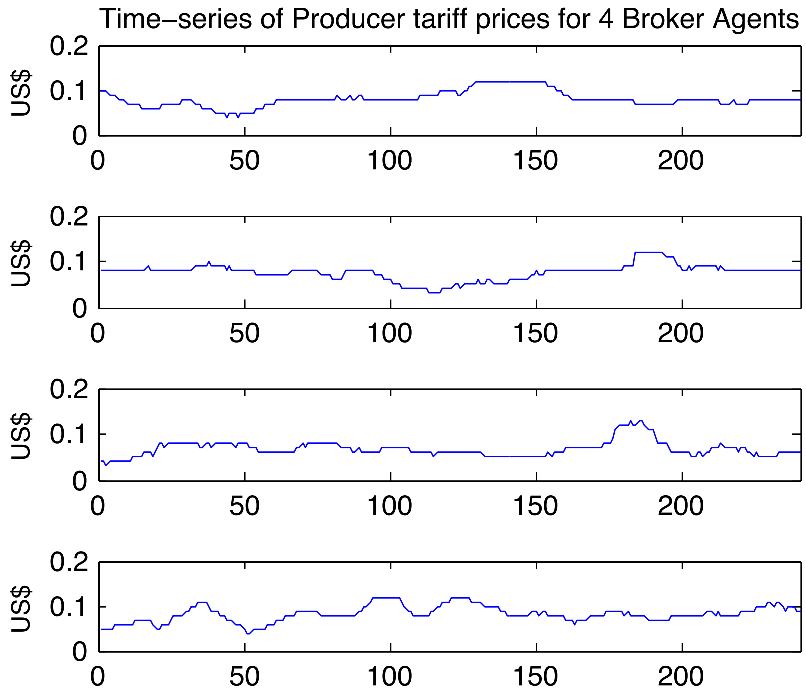


Figure 2.2: Four samples of producer tariff price sequences offered by broker agents employing a Fixed pricing strategy based on prices from Ontario IESO, a real wholesale market.

their prices. Some customers may not actively evaluate their available tariff options and therefore continue with their possibly suboptimal ranking. To capture this *inertia*, we take two steps:

1. If all the tariffs that a customer model evaluated at time $t-1$ are still offered at the same prices at t, then it simply returns the same ranking as in the previous time step; and
2. If the tariffs have changed, a customer model only considers switching to a different broker agent with a fixed probability, $q < 1$.

Moreover, some customers may choose tariffs with less favorable prices because other tariff attributes, such as the percentage of renewable energy or the lack of early termination penalties, may be preferable. So, each customer model ranks the price-ordered tariffs according to a discrete distribution, $\mathcal{X}$. For example, in an environment with 5 broker agents, $B1$ to $B5$, we have:

$$\mathcal{X} = \{x_k : \sum_k Pr(x_k = k) = 1, k = 1..5\} \tag{2.2}$$

With probability x_1, the customer model chooses the tariff with the best price; with probability x_2, it chooses the second best tariff, and so on.

2.2.2 Formulation and Strategy Learning

Let B_L be the *Learning* broker agent for which we develop an action policy using the framework of MDPs and reinforcement learning. The MDP for B_L is defined as:

$$M^{B_L} = \langle \mathcal{S}, \mathcal{A}, \mathrm{T}, \mathrm{R} \rangle$$

where:

- $\mathcal{S} = \{s_j : j = 1..|\mathcal{S}|\}$ is a set of states,

- $\mathcal{A} = \{a_j : j = 1..|\mathcal{A}|\}$ is a set of actions,

- $\mathrm{T}(s, a) \rightarrow s'$ is a transition function, and

- $\mathrm{R}(s, a)$ is a reward function.

$\pi : \mathcal{S} \rightarrow \mathcal{A}$ then defines an MDP action policy. Consider the example of Figure 2.2 again, which shows the producer tariff prices for 4 broker agents over 240 time steps. Assume that the Learning broker agent B_L is participating in a tariff market along with these 4 broker agents, B_1 to B_4. ($|\mathcal{B}| = 5$ in this example but the following analysis can be extended to any $|\mathcal{B}|$.)

2.2.2.1 Defining the State Space

A natural approach to representing the state space, $\mathcal{S}$, would be to capture two sets of features that are potentially important to how B_L would set its tariff prices:

1. the tariff prices offered by all the broker agents in the tariff market;

2. the number of consumers and producers in its current portfolio, Ψ^{B_L}.

Tariff prices are difficult to represent because prices in the real world are continuous over $\mathbb{R}^+$. We avoid the complexity of having to use function approximation methods by restricting the range of prices from 0.01 to 0.20, which represent a realistic range of prices in US dollars per kWh of electricity [DoE, 2010], and discretizing the prices in 0.01 increments to obtain 20 possible values for each tariff price.

With this simplification, if we were to model the Learning broker agent's MDP, M^{B_L}, to represent each combination of price values for 5 brokers at 2 tariff prices each, we would still have 20^{10}, or over 10 trillion, states in S to represent just the current tariff prices. To address this state explosion problem, we consider various statistics of the tariff prices such as the mean, variance, minimum and maximum prices for a given time t. However, since these statistics also vary over the valid price range, we would still have over 64 million states.

So, we apply the following heuristic to further reduce the state space. We define minimum and maximum producer and consumer tariff prices over the set of broker agents not including the Learning broker agent, B_L:

$$p_{t,C}^{min} = \min_{B_k \in \mathcal{B} \setminus \{B_L\}} p_{t,C}^{B_k} \tag{2.3}$$

$$p_{t,C}^{max} = \max_{B_k \in \mathcal{B} \setminus \{B_L\}} p_{t,C}^{B_k} \tag{2.4}$$

Figure 2.3 shows the minimum and maximum prices corresponding to the 4 producer tariff prices in Figure 2.2. We then introduce another simplification that drastically reduces the number of states. We define a derived price feature, *PriceRangeStatus*, whose values are enumerated as $\{$*Rational, Inverted*$\}$. The tariff market is *Rational* from B_L's perspective if:

$$p_{t,C}^{min} \geq p_{t,P}^{max} + \mu_L \tag{2.5}$$

where μ_L is a subjective value representing the margin required by B_L to be profitable in expectation. It is *Inverted* otherwise. We can now characterize the entire range of tariff prices offered by the other broker agents using just 4 states. Note that we do not discard the computed price statistics. We use their values in the implementation of some actions in $\mathcal{A}$ but we will not use them to discriminate the state space in $\mathcal{S}$; therefore our MDP policy does not depend on them.

We now address the second set of desired features in the state space; i.e., the number of consumers and producers in B_L's portfolio, $\Psi_t^{B_L}$. The number of consumers and producers can be any positive integer in $\mathbb{I}^+$ which if represented naïvely would result in a very large number of MDP states. We take a similar approach as above to reduce the state space by defining a *PortfolioStatus* feature that takes on a value from the set $\{$*Balanced, OverSupply, ShortSupply*$\}$.

In the final representation, the state space $\mathcal{S}$ is the set defined by all valid values of the elements in the following tuple:

$$\mathcal{S} = \langle PRS_{t-1}, PRS_t, PS_{t-1}, PS_t, \overrightarrow{p_t} \rangle$$

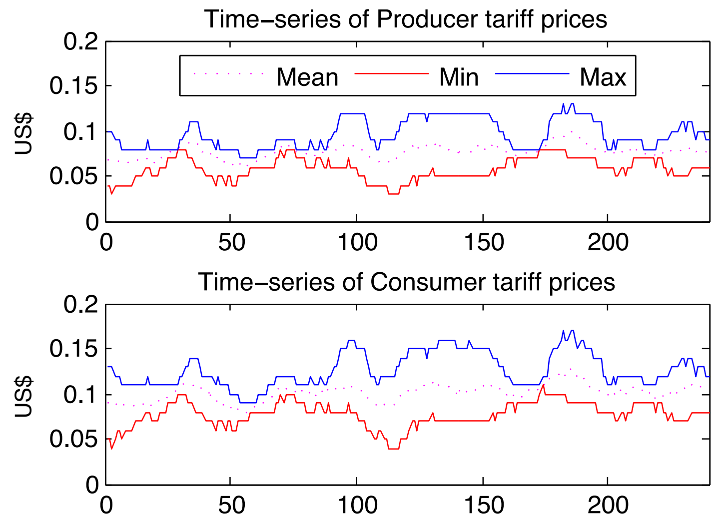


Figure 2.3: Minimum and maximum prices offered at each time step by the other broker agents, $\mathcal{B} \setminus B_L$, participating in the simulation.

where:

- PRS_{t-1} and PRS_t are the PriceRangeStatus values from B_L's perspective at $t-1$ and t,

- PS_{t-1} and PS_t are B_L's PortfolioStatus at time steps $t-1$ and t, and

- $\overrightarrow{p_t}$ is a vector of price statistics that are not used to discriminate the states for the MDP policy, but are included in the state tuple so that they can be used by the MDP actions, $\mathcal{A}$.

$$\langle p_{t,C}^{B_L}, p_{t,P}^{B_L}, p_{t,C}^{max}, p_{t,C}^{min}, p_{t,P}^{max}, p_{t,P}^{min} \rangle$$

We explicitly include PRS_{t-1} and PS_{t-1} to highlight states where the environment has just changed, so that the agent can learn to react to such changes quickly.

2.2.2.2 Defining the Action Space

Next, we define the set of MDP actions $\mathcal{A}$ as:

$$\mathcal{A} = \{Maintain, Lower, Raise, Revert, Inline, MinMax\}$$

where each of the enumerated actions represent a *tactic* defining how the Learning broker agent, B_L, sets the prices, $p^{B_L}_{t+1,C}$ and $p^{B_L}_{t+1,P}$ for the next time step, $t+1$. Specifically:

- *Maintain* publishes prices for time $t+1$ that are the same as those at t;
- *Lower* reduces both the consumer and producer tariff prices relative to their values at t by a constant, ς;
- *Raise* increases both the consumer and producer tariff prices relative to their values at t by a constant, ς;
- *Revert* increases or decreases each price by a constant, ς, towards the midpoint, $m_t = \lfloor \frac{1}{2}(p^{max}_{t,C} + p^{min}_{t,P}) \rfloor$;
- *Inline* sets the new consumer and producer prices as $p^{B_k}_{t+1,C} = \lceil m_t + \frac{\mu}{2} \rceil$ and $p^{B_k}_{t+1,P} = \lfloor m_t - \frac{\mu}{2} \rfloor$;
- *MinMax* sets the new consumer and producer prices as $p^{B_k}_{t+1,C} = p^{max}_{t,C}$ and $p^{B_k}_{t+1,P} = p^{min}_{t,P}$.

The transition function T is defined by numerous stochastic interactions within the simulator. The reward function R, unknown to the MDP, is simulated by the environment using Equation 2.1. Since this is a non-deterministic MDP formulation with unknown reward and transition functions, we use the Watkins-Dayan [Watkins and Dayan, 1992] Q-LEARNING update rule:

$$\hat{Q}_t(s,a) \leftarrow (1-\alpha_t)\hat{Q}_{t-1}(s,a) + \alpha_t[r_t + \gamma \max_{a'} \hat{Q}_{t-1}(s',a')] \tag{2.6}$$

where:

$$\alpha_t = 1/(1 + \mathit{visits}_t(s,a))$$

We vary the *exploration-exploitation* ratio to increase exploitation as we increase the number of visits to a state. When exploiting the learned policy, we randomly select one of the actions within 10% of the highest Q-value.

2.2.2.3 Experimental Results

We configured the simulation model described in Section 2.2.1.1 as follows. The load per consumer, κ, was set to 10kWh and the multiplicative factor for production capacity, ν, was also set to 10. The probability distribution $\mathcal{X}$ used to model customer preferences for ranking the price-ordered tariffs is fixed at $\{35, 30, 20, 10, 5\}$. (We study variations of this probability distribution in the next section.)

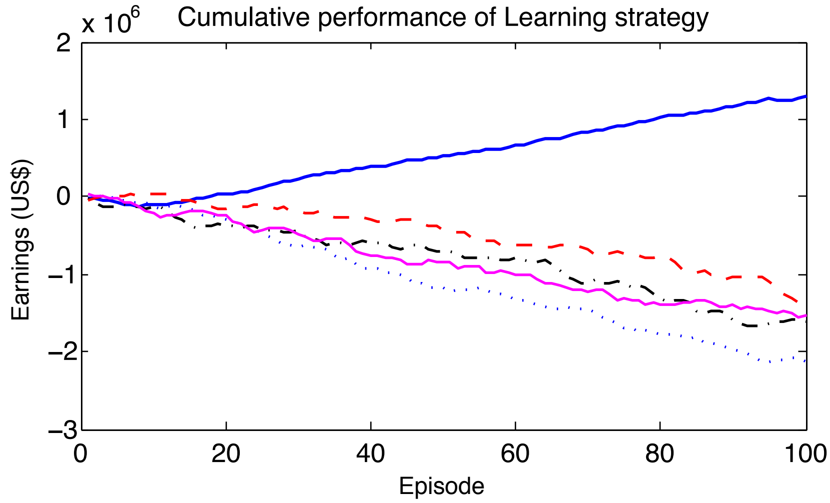


Figure 2.4: Cumulative earnings of the Learning strategy broker agent (upward trending line), relative to 4 data-driven broker agents.

The environment was initialized with 1000 consumers and 100 producers, so that supply and demand are balanced in aggregate. However, this does not result in a zero-sum game since all or some broker agents could be imbalanced even if the overall system is balanced. Since we do not model the wholesale market in this subset of the Smart Grid domain, broker agents cannot trade there to offset the balancing fees; it is therefore expected and observed that the average reward for most broker agents in our experiments is negative. The number of time steps per episode was fixed arbitrarily at 240; varying this number does not materially alter our results. When presenting aggregated results, we generally use runs of 100 episodes.

We learn an MDP policy, π, as the strategy for the Learning broker agent, B_L. Figure 2.4 shows the cumulative earnings of B_L compared to the earnings of 4 *Fixed* strategy data-driven broker agents, *i.e.*, their prices are fixed functions of averaged prices on the wholesale market. The plot clearly demonstrates the superior performance of the learned strategy compared to the fixed strategies of the data-driven broker agents.

We then consider how the learned strategy performs when compared to other effective strategies. For this evaluation, we use two hand-coded strategies presented in Algorithms 2.1 and 2.2. The *Balanced* strategy attempts to minimize supply-demand imbalance by raising both producer and consumer tariff prices when it sees excess demand and lowering prices when it sees excess supply. The *Greedy* strategy attempts to maximize profit by increasing its profit margin, *i.e.*, the difference between consumer and producer prices, whenever the market is Rational. Both strate-

Algorithm 2.1 BALANCED-STRATEGY(t, PS_t)

```
if PS_t = ShortSupply then
    a_{t+1} ← Raise
else
    if PS_t = OverSupply then
        a_{t+1} ← Lower
    end if
end if
```

Algorithm 2.2 GREEDY-STRATEGY(t, PRS_t)

```
if PRS_t = Rational then
    a_{t+1} ← MinMax
else
    a_{t+1} ← Inline
end if
```

gies can be characterized as *adaptive* since they react to market and portfolio conditions but they do not learn from the past.

Figure 2.5 compares the mean per-episode earnings and standard deviation of various strategies compared to those of 4 data-driven broker agents. The top-left subfigure shows the performance of a *Random* strategy (solid dot) where the broker agent simply picks one of the 6 actions in the action space $\mathcal{A}$ randomly. Its inferior performance indicates that the data-driven strategies used by the other broker agents are reasonably effective. The Balanced and Greedy strategies in the top-right and bottom-left subfigures respectively both show superior performance to the data-driven strategies. While they each achieve about the same average earnings, the Balanced strategy has much lower variance. The bottom-right subfigure shows the Learning broker agent's strategy, driven by its MDP policy, achieving higher average earnings than all other strategies, albeit with higher variance than the Balanced strategy.

While Figure 2.5 compares the strategies when played against Fixed data-driven strategies, Figure 2.6 shows the per-episode earnings of the various learning, adaptive and random strategies when played directly against each other. We see that the Learning strategy maintains its superior average earnings performance. The Balanced and Greedy strategies exhibit similar mean and variance properties as in Figure 2.5. Interestingly, the Random strategy now performs better than the Fixed data-driven strategy.

In a winner-take-all competitive setting, it is not enough to outperform the other strategies on average over many episodes. It is important to win each episode by having the highest earnings in that episode. Figure 2.7 shows the number of winning episodes for the Learning strategy in two

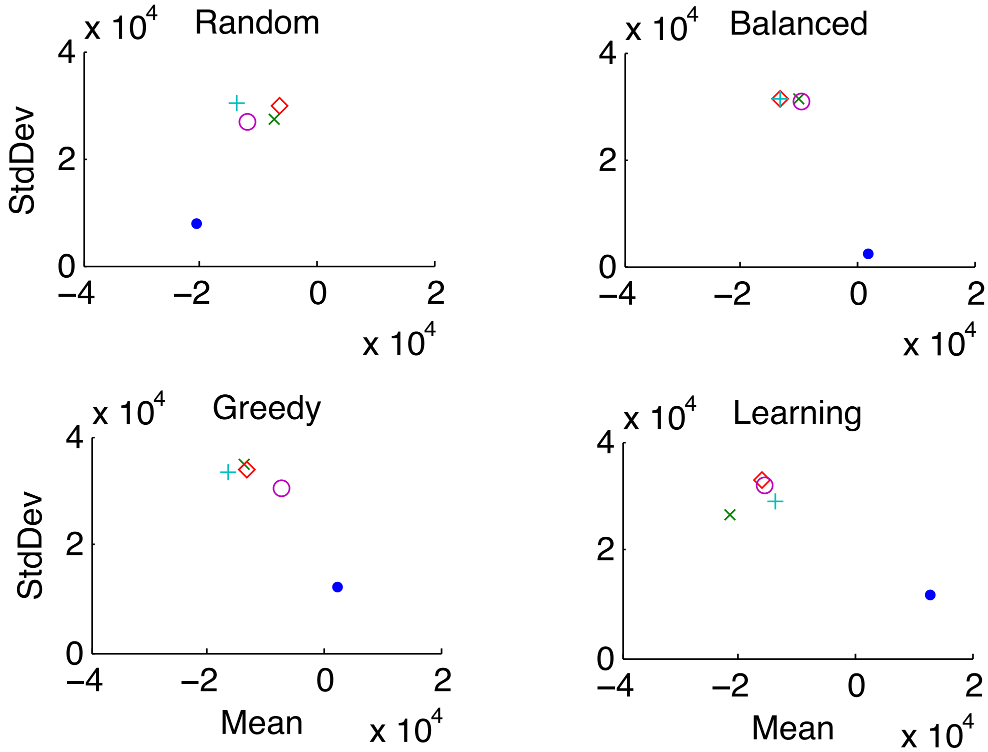


Figure 2.5: Each subfigure positions the mean and standard deviation of the labeled strategy (blue dot) relative to 4 data-driven Fixed strategy broker agents.

scenarios. The first set of dark-colored bars show that the Learned strategy wins about 45% of the episodes when playing against the Fixed data-driven strategies. The second set of bars show the results of playing the Learning strategy against the Fixed, Balanced, Greedy and Random strategies respectively. Remarkably, the Learning strategy now wins over 95% of the episodes.

We briefly address scalability in Figure 2.8, which shows the amount of time required to run 100-episode simulations with increasing numbers of broker agents. We expect typical tariff markets to include about 5 to 20 broker agents. We observe linear scaling with up to 50 broker agents, leading us to conclude that the MDP representation we have devised and the learning techniques we have employed remain computationally efficient in larger domains.

2.2.3 Equilibria with Multiple Learners

We have thus far shown that an autonomous broker agent can learn its strategy, using Markov decision processes (MDPs) and Q-LEARNING, and outperform other broker agents that use predetermined or randomized strategies. We now investigate the scenario in which multiple broker

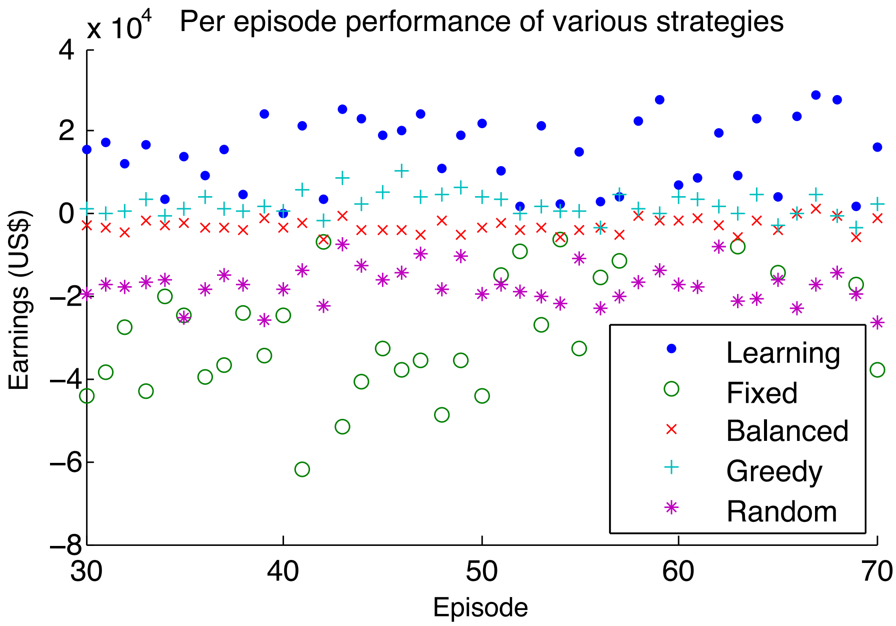


Figure 2.6: Comparison of cumulative episodic earnings for the various broker agent strategies played against each other simultaneously.

agents, not just one, are learning their strategies. We assume here that within the range of prices we consider, consumers may shift demand from one time to another but that their overall demand does not vary significantly. We then study the sensitivity of the learned strategies to specific learning parameters and also study the emergent attributes of the market prices and broker agent rewards. Specifically, we show that broker agents who employ periodic increases in exploration achieve higher rewards. Further, we find that different simulation models for how customers are allocated to broker agents, ranging from uniform distribution to market dominance by one or a few broker agents, result in remarkably distinct outcomes in market prices and aggregate broker agent rewards. The observed outcomes regarding broker agent rewards can be explained by economic principles of market-based competition.

Through our simulation experiments, we find that when many learning agents are participating in the tariff market, they each outperform the non-learning strategies. As an illustration of this set of experiments, Figure 2.9 shows the superior cumulative performance of two Learning broker agents, B_{L1} and B_{L2}, competing against broker agents using the Balanced and Greedy strategies. Note that the different broker agents are non-cooperative and we do not derive or analyze joint policies for the broker agents. Further note that the earnings summed over all broker agents are not expected to equal zero because of the balancing fee, ϕ_t, imposed by the distribution utility for portfolio imbalances. In an extended model of the Smart Grid domain that includes the

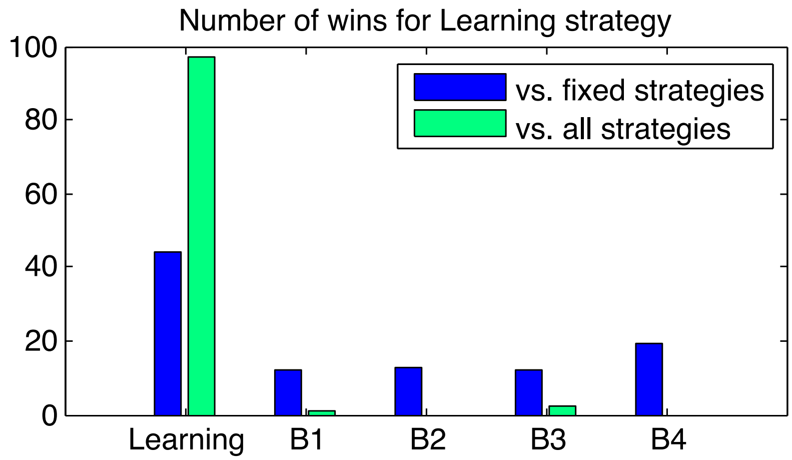


Figure 2.7: Number of *winning episodes* for the Learning strategy against Fixed strategy broker agents (blue bars) and mixed strategy broker agents (green bars).

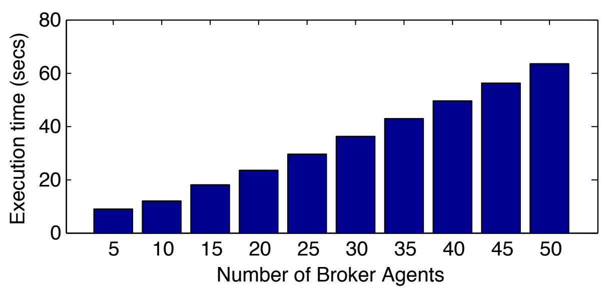


Figure 2.8: Growth of simulation execution time relative to the number of simulated broker agents.

wholesale market, broker agents would try and offset the potential balancing fees using forward trading contracts on the wholesale market [Ketter et al., 2010], as we describe in Section 2.3.

An important complexity in the multiple independent learners scenario is that learned policies must be updated over time as the other learners in the environment possibly update their own policies, thus leading to a non-stationary environment for each of the learners. We investigate the potential benefit of periodic relearning in this context, which leads to an additional broker agent strategy:

Relearning: This strategy builds upon the previously defined Learning strategy by modifying the exploration-exploitation tradeoff, which is typically a monotonic curve with exploration decreasing with time and across episodes. Our simulation model informs us that tariff prices are reset by each broker agent at the start of each episode to be within a fixed range of a configured parameter, p_0, *i.e.*, $p_0 \pm \varepsilon$. We hypothesize that a learning strategy might gain useful information by exploring to a greater extent at the beginning of each episode.

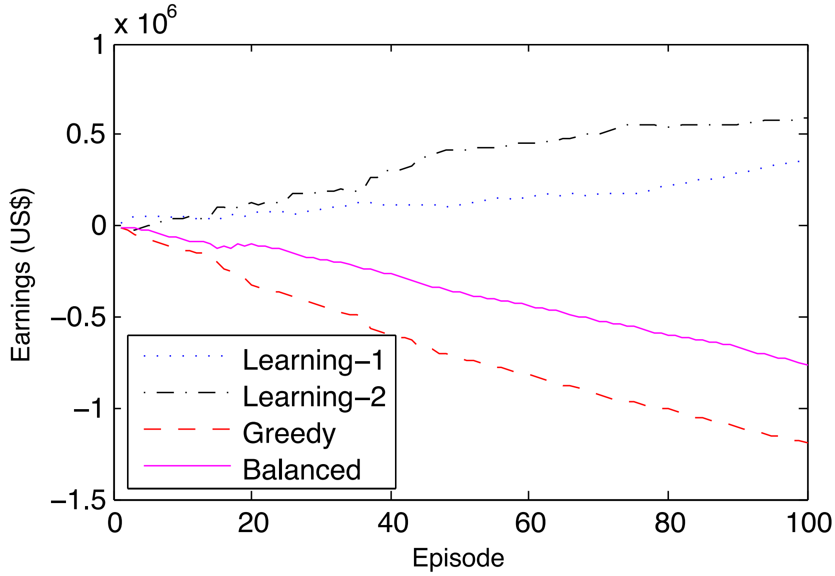


Figure 2.9: Cumulative per-episode earnings of two Learning strategy broker agents compared to two broker agents who use adaptive non-learning strategies.

We define a *relearning window*, w, as the number of time steps at the beginning of each episode where the MDP policy chooses a random action with a higher probability than it would have otherwise. Let ρ_f be a fixed exploration ratio and let ρ_t^c be the ratio implied by a monotonically decreasing curve at time t. At the beginning of a particular relearning window starting at t, the exploration ratio is set to $\max(\rho_f, \rho_t^c)$. After $w/2$ time steps, the ratio is changed to $\max(0.5\rho_f, \rho_{t+w/2}^c)$ and at the end of w time steps in the window, the ratio is restored to ρ_{t+w}^c. Figure 2.10 shows an exploration curve with relearning windows with w=40 for 10 episodes of 240 time steps each.

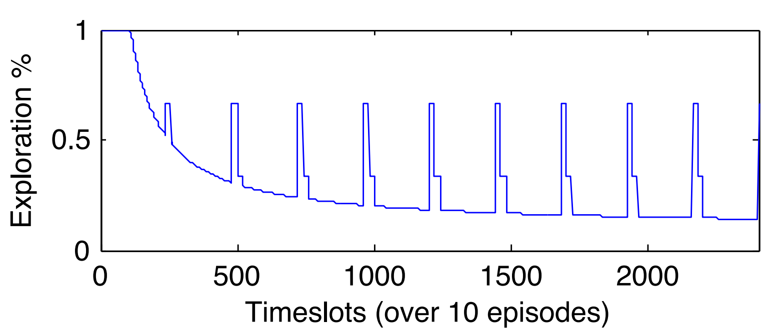


Figure 2.10: Modified exploration curve with *relearning* windows at the start of each of 10 episodes.

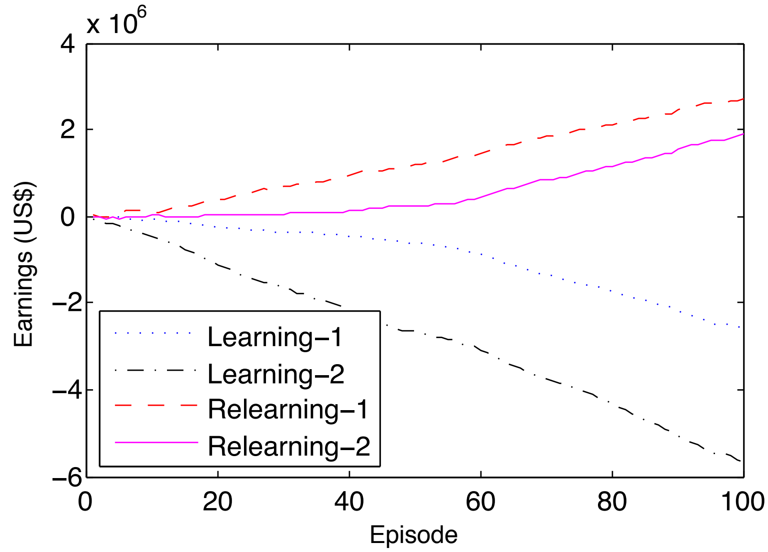


Figure 2.11: Cumulative per-episode earnings of two Relearning strategy broker agents compared to two Learning strategy broker agents.

As shown in Figure 2.11, we find that two broker agents who use such a *relearning exploration curve* achieve significantly higher rewards than two broker agents who use a monotonically decreasing curve.

Figure 2.12 shows the results of varying the relearning window. 4 broker agents of different window sizes compete with each other over 100 episodes. The y-axis represents a derived metric, *number of wins*, for evaluating broker agent performance; it counts the number of episodes where a given broker agent achieves the highest rewards for that episode. We find that increasing the

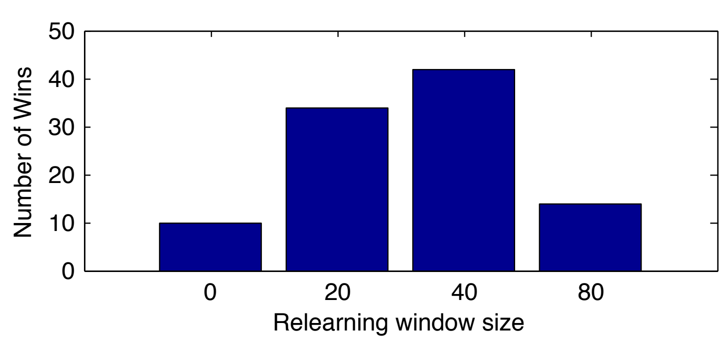


Figure 2.12: A relearning window size w of 40 time steps produces more wins than other window sizes.

window size helps up to a maximum and then hurts after that. Intuitively, this makes sense since a large relearning window decreases the opportunity to exploit the relearned policy.

2.2.4 Customer Allocation Models

The tariff market simulation model that we have developed allocates customers to broker agents based on the total order preference ranking by each customer model of the published tariffs at time t. A probability distribution $\mathcal{X}$ determines the likelihood of a customer choosing a particular broker agent's tariff. In this section, we study the effects of varying $\mathcal{X}$, the customer allocation model.

For example, in an environment with 4 broker agents, $B1$ to $B4$, we have $\mathcal{X}$ as a discrete distribution:

$$\mathcal{X} = \{x_k : \sum_k Pr(x_k = k) = 1, k = 1..4\} \tag{2.7}$$

With probability x_i, a given customer model prefers the tariff with the ith most favorable price and thus the corresponding broker agent. The possible distribution values for $\mathcal{X}$ are infinite, but we analyze 4 distinct and interpretable instances for the 4 broker agents scenario:

- *Uniform* allocates customers to broker agents evenly. In this model, broker agents have no incentive to publish tariffs that customers would find to be preferable because they acquire customers with the same probability regardless of their published tariff prices.

$$\mathcal{X} = \{0.25, 0.25, 0.25, 0.25\} \tag{2.8}$$

- *Biased* is more likely, compared to Uniform, to allocate customers to their preferred broker agents.

$$\mathcal{X} = \{0.50, 0.25, 0.15, 0.10\} \tag{2.9}$$

- *Volatile* allocates each customer to any of the broker agents other than the one least preferred by that customer. Viewed from a broker agent's perspective, this is a volatile allocation model because it severely penalizes a broker agent for publishing tariffs that may not be preferred by any customers.

$$\mathcal{X} = \{0.33, 0.33, 0.33, 0.01\} \tag{2.10}$$

- *Dominant* allocates most customers to the broker agent with the most desirable tariffs. This is also a volatile allocation model but it provides a large advantage to a single broker agent instead.

$$\mathcal{X} = \{0.85, 0.05, 0.05, 0.05\} \tag{2.11}$$

To understand the impact of these customer allocation models, we first study the market prices that emerge from the interaction of 4 broker agents, each independently using the Learning strategy, under each allocation model.

In Figure 2.13, the data points in the subplot for each customer allocation model represent a consumer tariff price offered by any of the 4 broker agents at a given time t over 30 episodes. So, for each of the 240 time steps per episode, *i.e.*, each value on the x-axis, there are 4 $\times$ 30 = 120 data points along the y-axis.

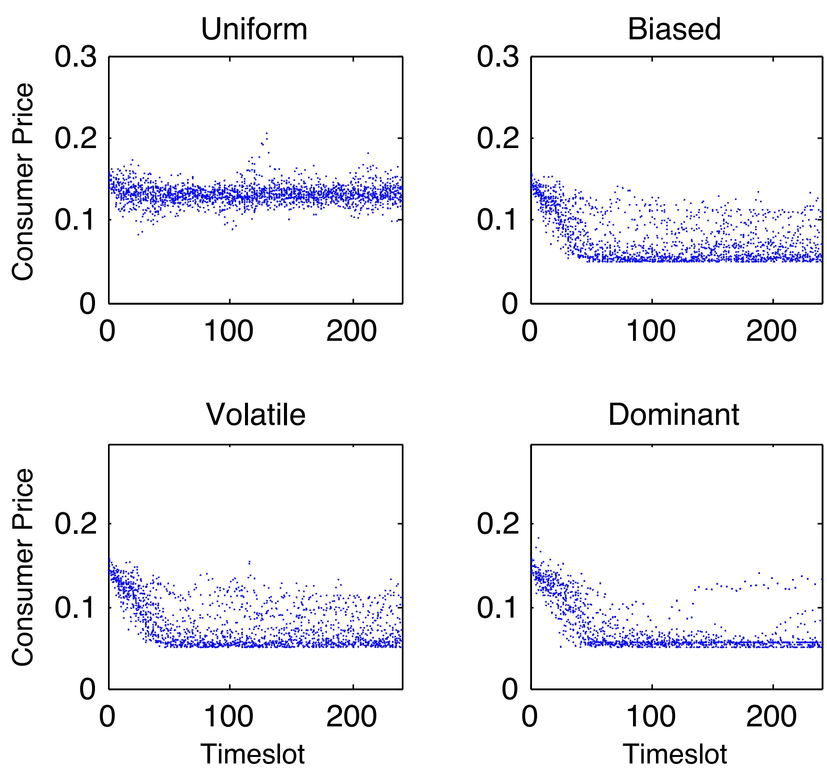


Figure 2.13: Customer tariff price evolution over an episode, overplotted for 30 episodes, for each of the labeled customer allocation models.

At the start of each episode, each broker agent publishes a consumer tariff price in the range $p_0 \pm \varepsilon$, where p_0 is a configured parameter to the simulation model. Given that each broker agent is acting independently, we might expect the published prices to diverge over the full range of allowed consumer tariff prices, 0.04 to 0.30. Such divergence would show the 120 data points at each x-value spread out over the y-axis, especially for the later time steps in each episode. But remarkably, we instead see a very high concentration of prices. This is probably explained by the learning behavior of each broker agent. Depending on the customer allocation model applied, each broker agent independently converges to the same policy or each learns a policy that keeps its published prices in very close proximity to the prices published by the other broker agents.

For models other than Uniform, there is a pronounced tendency of the broker agents to drive prices downwards very rapidly. The Biased and Volatile models result in prices with more variance in earlier episodes and as learning progresses, they converge to lower prices. With the Dominant model, prices converge to the lower limit within the first 3-4 episodes. Lower consumer prices are desirable but we cannot conclude that the Dominant model is preferable since producer prices, which are not shown here, also converge to their lower limit—when producers are faced with low prices they may be forced to withdraw from the market if they cannot reduce costs sufficiently to remain profitable. Such an outcome would defeat the goal of encouraging increased participation from distributed small-scale power producers.

The bar plot in Figure 2.14 shows typical cumulative earnings over 100 episodes for each of the 4 Learning broker agents competing under the 4 customer allocation models. The first group of bars show cumulative earnings for each broker agent under the Uniform model; we see that all broker agents are highly profitable with an approximately equal share of the earnings. The Biased model, compared to the Uniform model, has lower sum earnings over all broker agents and also shows more variance amongst their earnings. The Volatile model further reduces the sum earnings with about the same variance as the Biased model. The Dominant model stands out for the highly negative earnings of some of the broker agents.

Even though all of the broker agents are using the same Learning strategy, the policies they learn and their cumulative performance can be quite different depending on two factors: (i) the stochastic prices that they publish at the beginning of each episode, and (ii) the customer allocation model being applied. For example, a broker agent with initial consumer prices that are not preferred by many customers may not be able to build a sufficiently large or balanced portfolio of customers if the customer allocation model has a very low customer allocation probability, x_i, for less preferable tariffs. This can cause the broker agent to learn higher Q-values for an action like MinMax because that action would increase revenue from the existing portfolio of customers. However, such an action is likely to further alienate customers by making the published prices

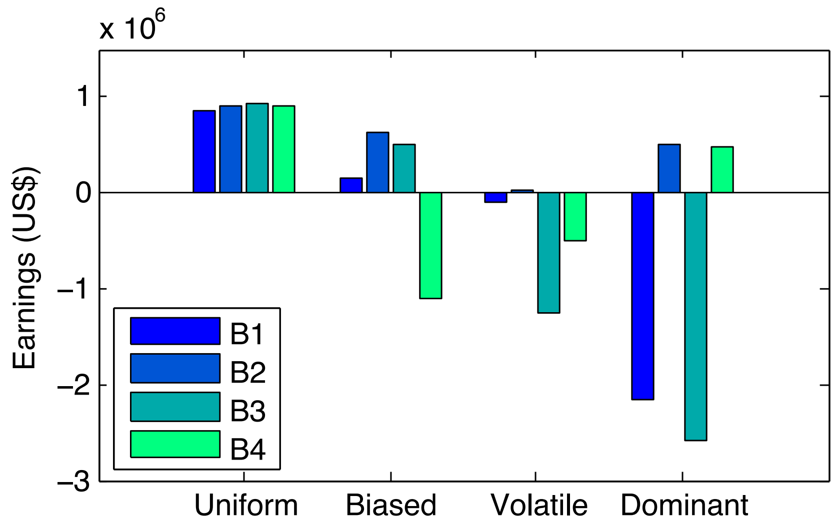


Figure 2.14: Earnings for 4 Learning strategy broker agents, B1 to B4, played against each other under the labels customer allocation models.

even less preferable. This negative effect carries forward into subsequent episodes, even if the prices are reset at each episode, because the learned Q-values, if not unlearned quickly, continue to influence which actions are chosen by the broker agent. Therefore, the initial conditions and the customer allocation model can be quite influential in distinguishing the cumulative performance of the broker agents even when they all use the same Learning strategy.

An interpretation of the results in Figure 2.14 is that the Biased model is attractive because it yields positive earnings for a majority of the players and penalizes a few that publish undesirable tariffs. The sum earnings over all broker agents is lower compared to the Uniform model which may be due to more economic value accruing to the customers instead of the broker agents. In the Volatile and Dominant models, broker agents have a greater probability of building imbalanced portfolios by acquiring a large market share of consumers but not producers or vice versa. Such imbalances are likely to cause large negative earnings for the broker agents, possibly forcing some of them to exit the market. If a number of broker agents leave the market, market power is concentrated in the initially successful broker agents, possibly reinforcing a Dominant allocation model and leading to an inefficient monopoly over time. This interpretation of the Dominant customer allocation model seems consistent with established economic principles, which maintain that natural monopolies can arise due to market-based reinforcement and asymmetric growth of one market participant, at the cost of the others, even though each participant has similar capabilities [Pindyck and Rubinfeld, 2004].

2.3 Price Prediction in Wholesale Markets

The term $|\kappa\Psi_{t,C} - \nu\kappa\Psi_{t,P}|$ in Equation 2.1 represents the supply-demand imbalance in a broker agent's portfolio at time t. This imbalance is penalized using a substantial balancing fee, ϕ_t. The broker agent is therefore motivated to offset the anticipated imbalance in his portfolio by buying or selling forward contracts in the wholesale market. Conversely, if prices in the wholesale market are expected to be unfavorable, the broker agent may choose to try and alter the makeup of their portfolio by changing their published tariffs or exercising optional controls that limit supply or demand from their portfolio. It is therefore critical that a broker agent be able to understand the evolution of prices in the wholesale market and be able to predict them.

In this section, we present an analysis of the evolution of hourly electricity prices in a modern wholesale electricity market with the goal of predicting hourly forward prices for at least the next 24 hours so that a broker agent can effectively manage trading risk. We base our analysis on real market data from the Ontario Independent Electricity System Operator (IESO) from 2002 to 2011 and corresponding real weather data from the US National Climatic Data Center (NCDC). We study the overall characteristics of the prices using density estimation and k-means clustering. We then restrict our study to the year 2009 and apply regression and multi-class classification methods to estimate the changes in hourly prices based on a number of market- and weather-related covariates. We find a strong correlation of prices with historical prices, so we also extend the study to a time series analysis of only the prices. Our analysis shows that a combination of the multi-class classification approach and the multiplicative seasonal ARIMA model from the time series analysis can be used to predict the hourly forward prices with confidence.

Trading in the IESO market is conducted using a periodic double auction mechanism that is cleared once every hour throughout each day. During a given hour, trading is allowed in electricity that is intended to be consumed during the next H hours (typically, H=24.) So, in fact, the wholesale market conducts H simultaneous auctions to determine the clearing price for each of the H future time steps.

Let $\eta_{t,t'}$ be the wholesale market's clearing price at a given time t for a future trading time t'. Let $q^{B_k}_{b,t,t'}$ and $q^{B_k}_{s,t,t'}$ be the quantity of electricity bought and sold respectively by broker agent B_k at time t for each of the open trading hours t'. Equation 2.1 is then extended to include the trading costs as:

$$r_t^{B_k} = p_{t,C}^{B_k}\,\kappa\,\Psi_{t,C} - p_{t,P}^{B_k}\,\nu\,\kappa\,\Psi_{t,P} + \underbrace{\sum_{t<t'\leq t+H} \eta_{t,t'}\,(q^{B_k}_{s,t,t'} - q^{B_k}_{b,t,t'})}_{\text{trading costs}} - \phi_t\,|\kappa\,\Psi_{t,C} - \nu\,\kappa\,\Psi_{t,P}| \quad (2.12)$$

and the subgoal of the broker agent is to predict $\eta_{t,t'}$ under various market scenarios.

2.3.1 Analysis of Historical Prices

The market data from the IESO and corresponding weather data from the NCDC contain several attributes relevant to our analysis:

1. the clearing prices, officially called Hourly Ontario Electricity Prices (HOEP),
2. hourly electricity demand,
3. hourly wind power production,
4. daily operating reserve prices,
5. hourly uplift settlement charges, and
6. hourly temperatures and wind speeds from various cities across Ontario.

Samples from this dataset for the year 2009 are shown in Figure 2.15. Over the past decade, the electricity markets have been going through significant restructuring. So we first attempt to characterize the prices from 2002 to 2010 to see if we can find evidence of the restructuring. We take median daily prices over this period and apply k-means clustering. Figure 2.16 shows the resulting clustering. We see some evidence of higher price volatility in the early years, but more notably, there is a significant downward shift in prices over the last 2-3 years.

We further characterize the range of prices using a kernel density estimate of the median daily prices computed using the hourly Ontario electricity prices as shown in Figure 2.17. We observe that the median daily prices are vastly skewed—and the skew is even greater in the raw hourly prices, not shown. Also note the negative prices, which reflect scenarios where at times of unusually high and unexpected power supply (*e.g.*, production from wind turbines during a storm), sellers will *pay* to have buyers consume the electricity they generate so that they don't have to try and store the energy. The negative prices make it difficult to apply typical transformations that generate normal distributions, *e.g.*, taking the log of the prices to adjust for the skew.

We restrict our subsequent analysis to the more stable price regime of hourly prices for 2009, which gives us 8760 samples. In Figure 2.18, we plot correlations for a subset of data features against the changes in hourly prices. The top-left plot in Figure 2.18 shows significant autocorrelation of the hourly price change with the hourly price change in the previous hour. We also note that Ontario demand changes and Toronto temperatures have some positive and negative correlations with historical HOEP. However, if we include all of these covariates in a multivariate

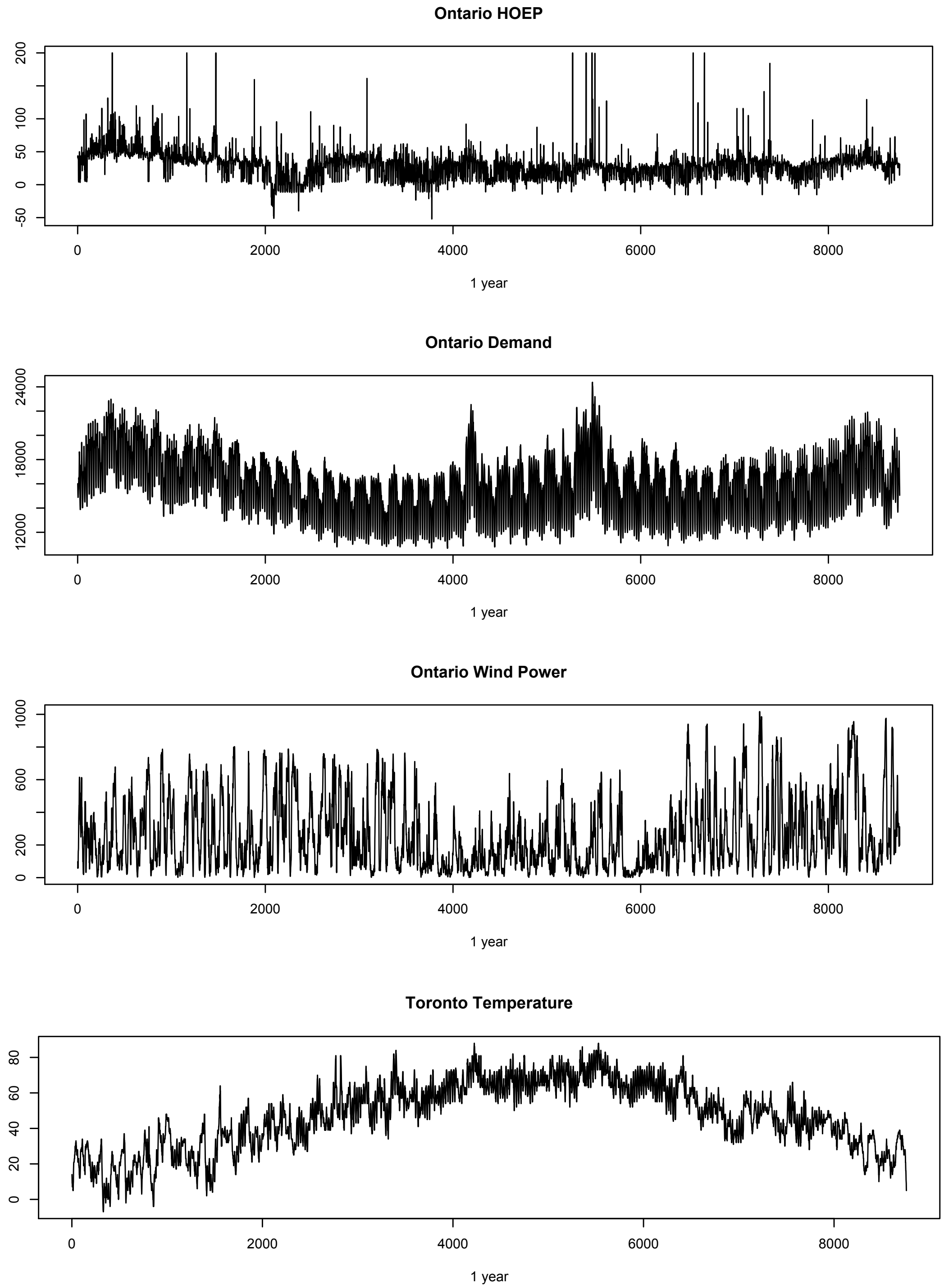


Figure 2.15: Samples of raw data, hourly for all of 2009, used in multivariate regression for prediction of hourly Ontario electricity prices (HOEP).

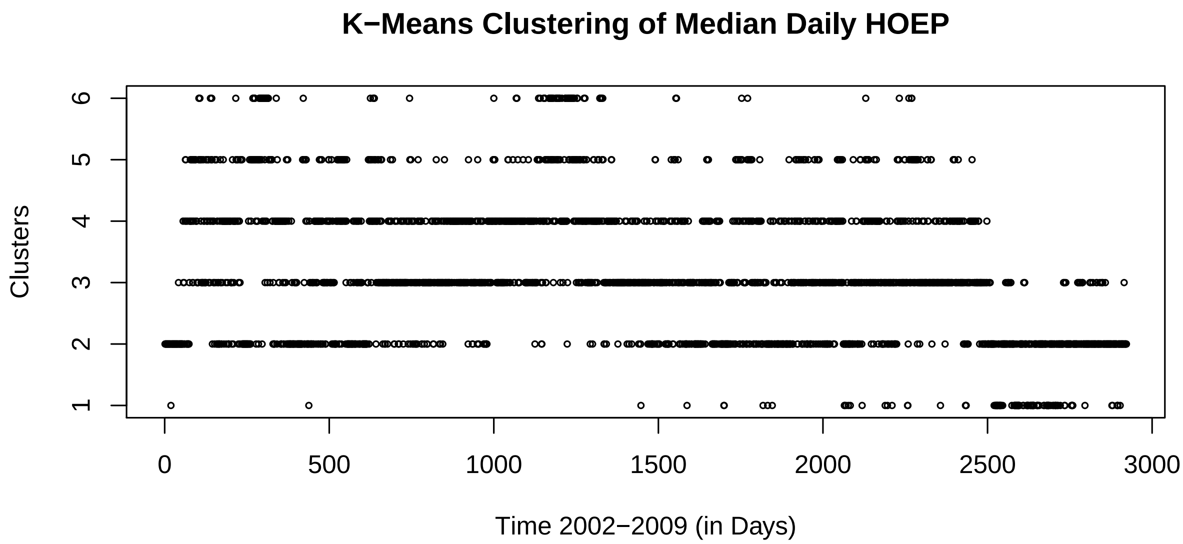


Figure 2.16: K-means clustering of median daily prices computed from hourly Ontario electricity prices (HOEP) from 2002 to 2009.

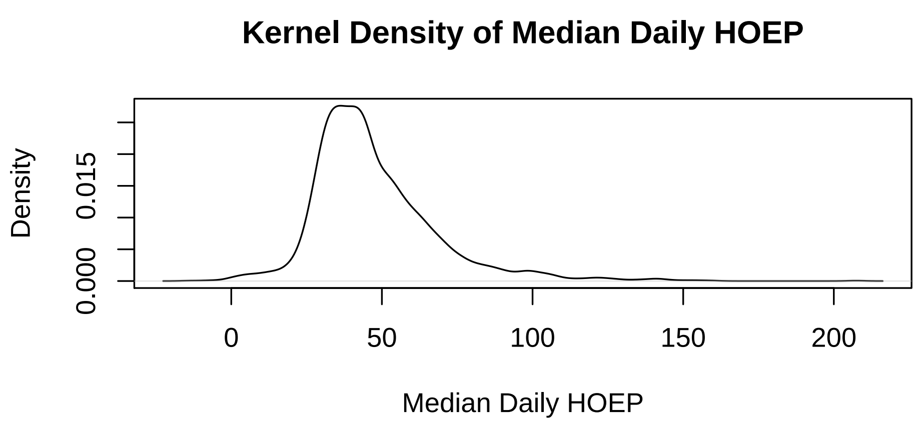


Figure 2.17: We plot the kernel density of the median daily prices to characterize the skew in the price distribution and explore options for transformation to normality.

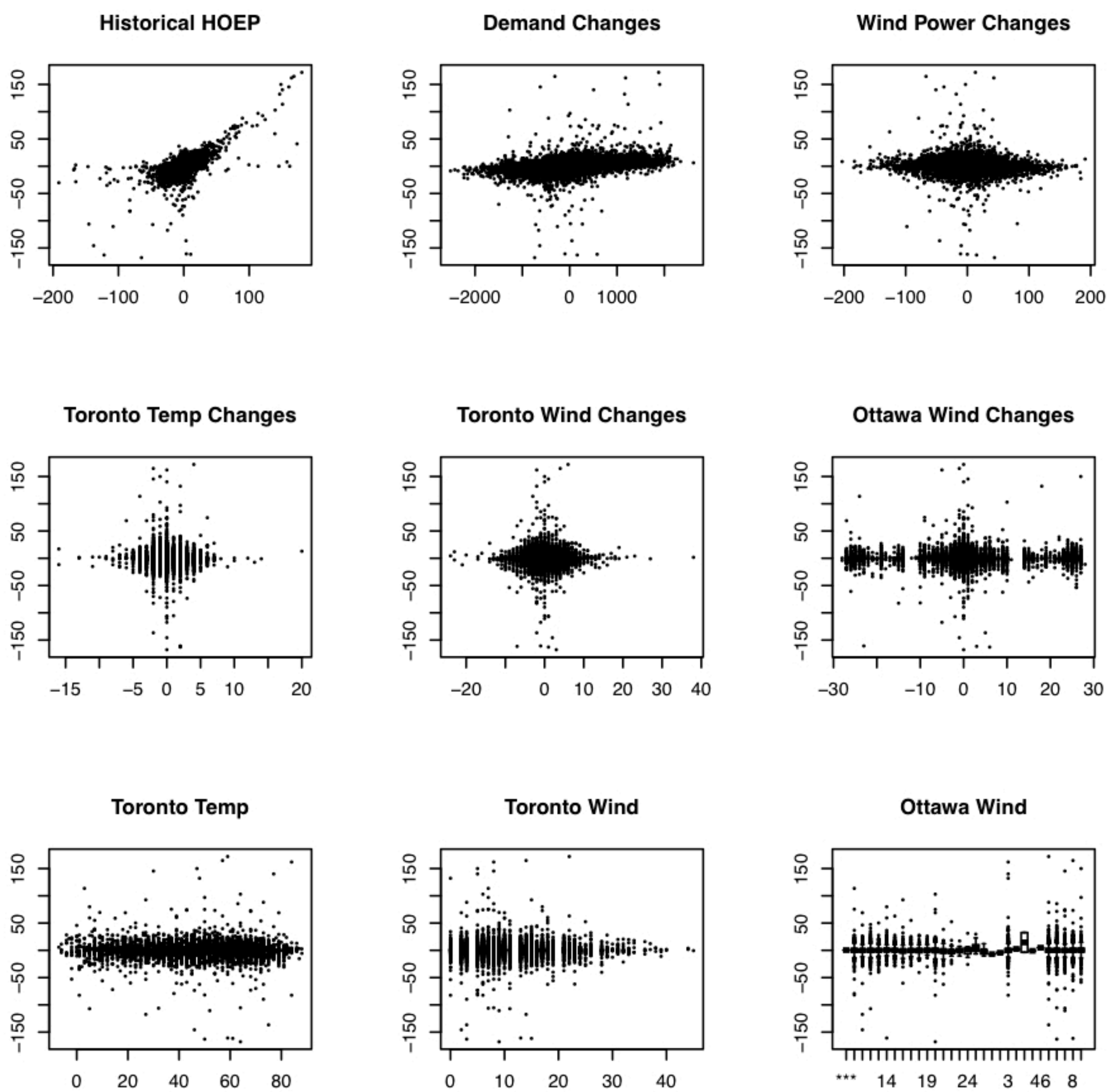


Figure 2.18: Correlation plots for various covariates, which include raw values of data described in Figure 2.15 and their hourly changes, considered for regression.

regression, the effect of demand changes and Toronto temperature is subsumed by the predictive power of historical HOEP.

2.3.2 Classification of Price Changes

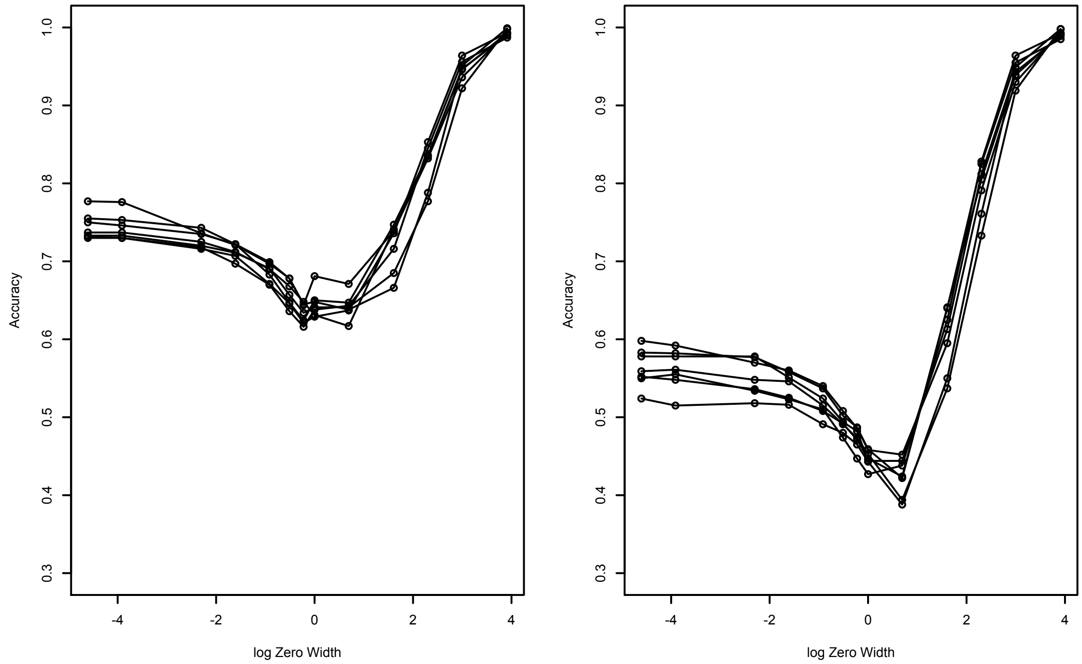
Since multivariate regression yielded limited success, we turn our attention to the more tractable problem of classification. Specifically, we aim for 3 target classes which would help inform the strategy of a broker agent in our problem domain. We label the classes as follows:

$$h(e_t) = \begin{cases} -1 & \text{if } e_t < \delta \\ +1 & \text{if } e_t > \delta \\ 0 & \text{otherwise} \end{cases} \tag{2.13}$$

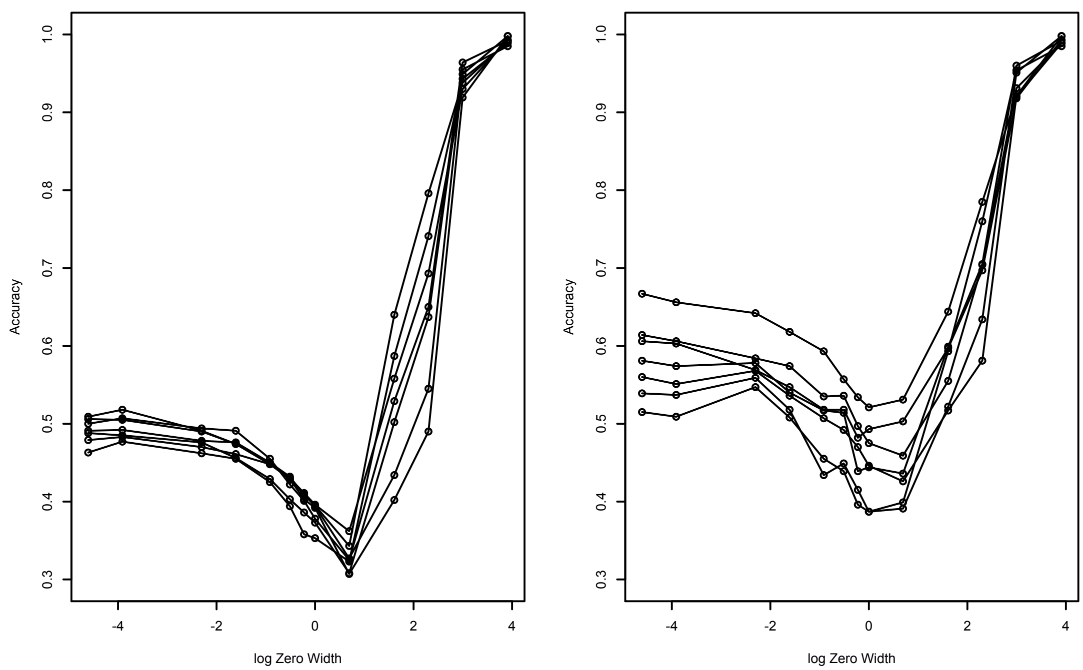
where e_t are hourly price changes and δ is a threshold parameter that determines how much of a deviation from zero would be considered *no change* by the broker agent. This is an intuitive classification because minor price changes are common in real-time markets and are difficult to predict or explain, so they are not worth worrying about. The +1 and -1 labels focus the broker agent's attention on anomalous price predictions that likely require explicit action from the agent. For example, broker agents typically acquire some *controllable capacities* as part of their portfolio. Such capacities, *e.g.*, a household water heater, can be explicitly shut off by the service operator in response to the broker agent exercising its option to do so. When a broker agent determines that the price movement in the wholesale market is sufficiently unfavorable, it can include the exercise of such options in its overall strategy.

Under this classification setup, we perform several experiments using a one-vs-all 10-fold cross-validated multi-class SVM with radial basis kernels. Figure 2.19 shows some results. The x-axis of each subfigure represents *log* δ, ranging from -4 to +4, and the y-axis reports classification accuracy ranging from 30% to 100%. The SVM in the first subfigure only considers historical price changes as features, the second subfigure considers other market-based features, the third considers weather-based features and the fourth all of the features together. Each line in each subfigure represents the performance of the classifier on an entirely different test set. The combinations of features again confirms what we found in the correlation plots and regression– that historical prices are indeed the best indicator of future prices in the very short term future.

Note the distinctive *black swan* pattern in each subfigure. We find that classification accuracy dips drastically when $0 < \textit{log } \delta < 1$. These dips show that price changes up to about 3 CAD/MWh are difficult to classify, although using just the historical prices, as in the first subfigure, we can achieve accuracy of about 65% even in this worst case scenario. With higher vartheta



(a) Historical HOEP changes (left), and demand and wind power production changes (right).



(b) Toronto temperature and windspeed and Ottawa windspeed (left), and all features (right).

Figure 2.19: Classification accuracy for different δ (Eq. 2.13) for the captioned feature combinations.

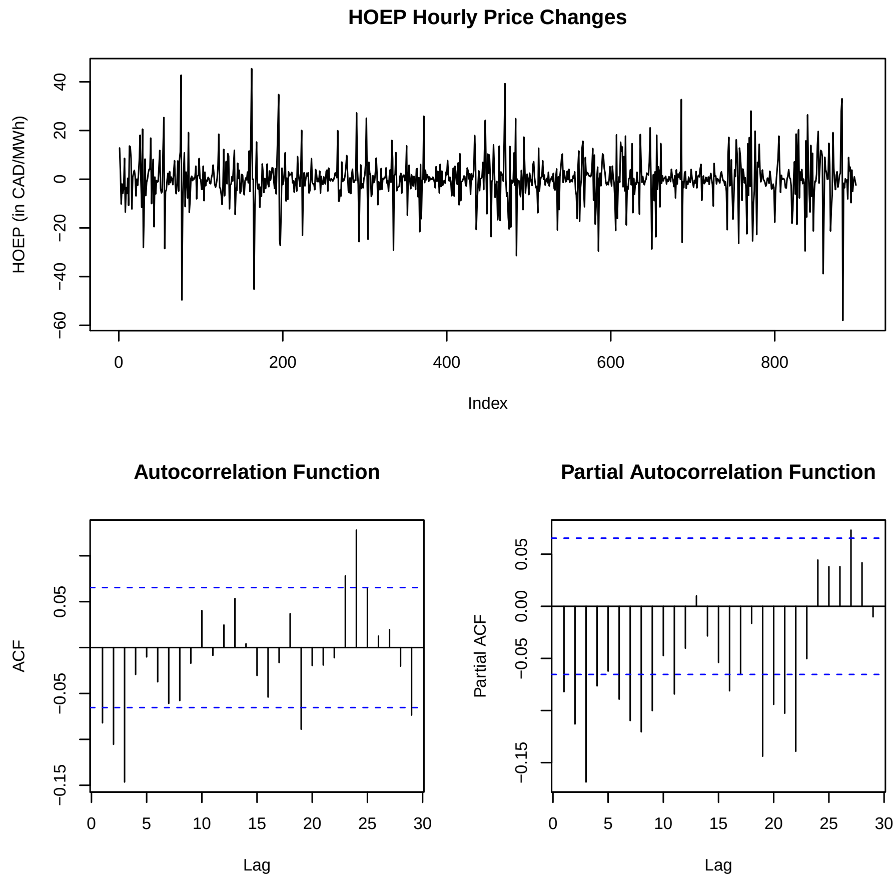


Figure 2.20: Time series of hourly changes in 2009 HOEP and its ACF/PACF diagnostic functions.

values, the classification accuracy climbs to full accuracy as would be expected. Therefore, if we use the classification only to identify the +1 and -1 labels for "large" price changes, we can achieve arbitrary classification accuracy by increasing the δ threshold. A suitable value for δ might be the one that gives a 95% confidence in the classification.

Given that the SVM-based classifier helps identify the direction of the larger price changes, *i.e.*, the outliers, we now focus on modeling the more typical price changes. We pursue a time series analysis towards this goal. Figure 2.20 shows the 2009 hourly price changes and the corresponding autocorrelation and partial autocorrelation functions (see Appendix C).

After some analysis we find that the following ARIMA $(0, 1, 3) \times (0, 1, 1)_{24}$ multiplicative seasonal model fits fairly well (see Appendix C). Y_t is the hourly price at time t and e_t is the innovation or deviation at time t:

$$Y_t = Y_{t-1} + Y_{t-24} - Y_{t-25} + e_t + \varepsilon_t \tag{2.14}$$

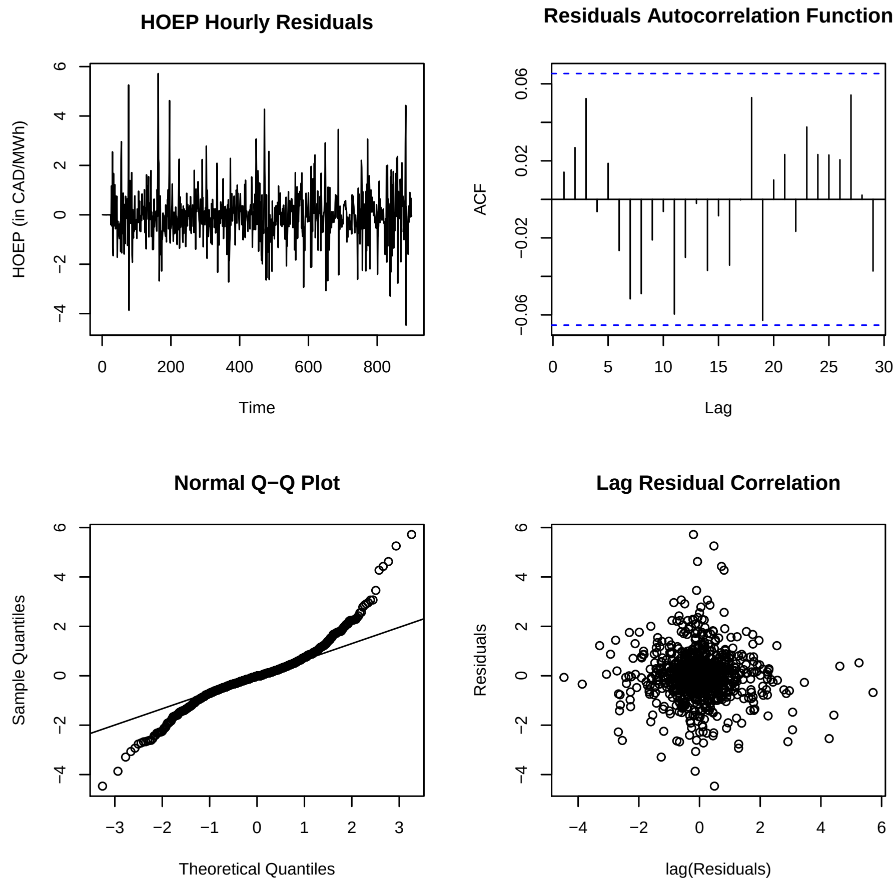


Figure 2.21: Residuals from multiplicative seasonal ARIMA model fit on 2009 HOEP.

where:

- $\varepsilon_t = \theta_1 e_{t-1} + \theta_2 e_{t-2} + \theta_3 e_{t-3} + \Theta_1 e_{t-24} + \theta_1 \Theta_1 e_{t-25} + \theta_2 \Theta_1 e_{t-26} + \theta_3 \Theta_1 e_{t-27}$
- $\theta_1 = -0.3034, \theta_2 = -0.2311, \theta_3 = -0.2200$
- $\Theta_1 = -0.9224$

The model does a good job of explaining all the complex correlations apparent in Figure 2.20 and the seasonal component is logically sound because we know that we are analyzing hourly prices. However, the heavy tails in the original price series, and correspondingly in the residuals, e_t, are certainly a cause for concern. The theory of ARIMA models is based on a multivariate Gaussian assumption on the innovations, e_t, at each time step [Cryer and Chan, 2008]. Our residuals seem to be better represented by a skewed Cauchy distribution but the theory based on Cauchy distributions is extremely difficult in such massive multivariate models. An adapted

ARIMA model based on a t-distribution might also be a way to address the observed thick-tailed distribution. Conversely, we could imagine somehow thinning the tails by removing some of the outliers that we expect to be able to classify away using our SVM and then shifting and transforming the data to address the negative prices and the skewness. However, we are assuming that the spikes in the residuals are due to thick tails and not due to changes in variance, *i.e.*, heteroscedasticity.[1]

The plots for analyzing the residuals are shown in Figure 2.21. We note that the residuals at first appear to have non-constant variance but if we look at the Q-Q plot, we can see that the tails of the distribution are significantly heavier than would be expected of normally distributed errors; therefore the periods of apparent increased variance are likely just many occurrences of these *outliers*. The other two plots yield much more positive results in that we see no significant remaining autocorrelations in the residuals. We can therefore reasonably conclude that the proposed ARIMA model is a good fit when prices stay within a reasonable range, but we should not trust the model to predict the relatively frequent occurrences of larger price changes. The seasonal period of 24 hours makes intuitive sense since these are hourly prices and we expect correlations for the same hour from one day to the next.

The resulting forecasting model, illustrated for 72 hours in Figure 2.22 shows the periodic nature of the price evolution. This ARIMA forecasting model can be combined with the outlier classification model as part of a broker agent's price prediction strategy.

2.4 Chapter Summary

In this chapter, we explored the problem of developing pricing strategies for broker agents in Smart Grid tariff markets. We formalized the tariff market domain representation and the goal of a broker agent. We contributed a scalable MDP formulation including a set of independently applicable pricing tactics. We demonstrated the learning of an effective strategy without any prior knowledge about the value of available actions. We evaluated the learned strategy against non-learning adaptive strategies and found that it almost always obtains the highest rewards. We showed that multiple broker agents using learned strategies each outperform non-learning broker agents. We contributed a non-monotonic exploration heuristic for *relearning* to account for changes in other broker agents' strategies over time. This heuristic is designed for environments with periodic changes. We demonstrated, using simulation-based experiments, that broker agents who use this relearning heuristic achieve higher rewards. These results demonstrate that rein-

[1] If indeed the series has non-constant variance, we would need to add a GARCH model to the innovations and look for correlations in volatility instead of simply assuming that the innovations are Gaussian white noise.

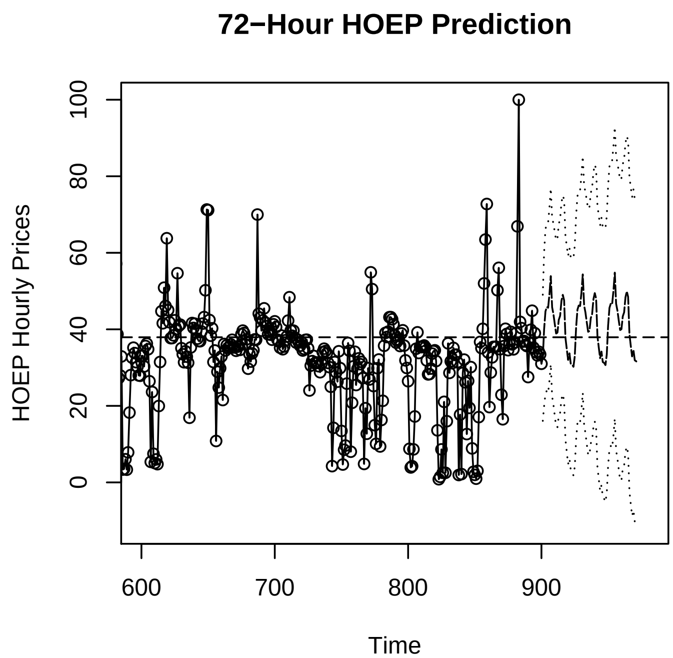


Figure 2.22: Typical forecast for the next 72 hours based on the ARIMA model of Eq. 2.14.

forcement learning with domain-specific state aggregation techniques can be an effective tool in the development of autonomous broker agents for Smart Grid tariff markets. We also contributed an analysis of the behaviors resulting from the interaction of multiple learning strategies in the tariff market. Specifically, we found that market prices are driven downwards rapidly and we found that the emergent aggregate broker agent rewards are largely consistent with economic principles, thus validating our simulation approach. These results can provide guidance for the design of Smart Grid tariff markets in the real world.

We also analyzed real price data from a representative wholesale electricity market along with corresponding weather data to build a prediction model for hourly price changes for over 24 hours into the future. This prediction model uses a multiplicative seasonal ARIMA model for usual price patterns and a 3-label SVM-based classification model to predict the likelihood of larger price changes in the positive or negative direction. While the magnitude of these larger changes cannot be predicted by the SVM, we claim that such predictions are difficult to model without more knowledge of exogenous non-repeating factors that may be causing those spikes.

Our work presented in this chapter forms some of the earliest research on formulating and tackling the problem of broker agent strategies. Some of our results were first published at IJCAI-11 [Reddy and Veloso, 2011c] and AAAI-11 [Reddy and Veloso, 2011a]. Much more

remains to be done in increasing the sophistication of problem representations and strategy learning. Our approach and results have since been used as a basis for further research on broker agent strategies, *e.g.*, function approximation and SARSA-based learning are added to the broker agent's learning strategy in [Peters et al., 2013]. While such welcome advancements continue the line of research introduced in this chapter, we turn our focus to the challenges of simulating the larger tariff market domain and developing strategies for *customer agents*. Nonetheless, some of the techniques that we develop in the context of customer agents in later chapters can also be applied to the decision-making challenges of broker agents.

Chapter 3

Customer Model Simulation

Smart Grid tariff markets are yet to be implemented in much of the real world, and where they exist they are nascent. So, we rely on agent-based simulation to develop and validate the contributions of this thesis, modeling future markets and agent behaviors using real world data where possible. In the development of such an environment, we encounter the problem of time series simulation based on prior sample data, online bootstrap data, and subjective biases that must be introduced to simulate specific behaviors. We address this problem using a novel Bayesian time series simulation method, which we describe in this chapter following a brief overview of the overall simulation environment.

3.1 The Power TAC Environment

While significant thought and research has already been expended on techniques for renovating our power grids into a *Smart Grid*, that effort has largely been focused on infrastructure for reliability and maintainability [United States Department of Energy, 2012]. Advanced metering infrastructure (AMI) components such as *smart meters* allow consumers to monitor electricity usage in real time and better manage their consumption patterns. However, innovations beyond that technical foundation are needed.

New market structures can potentially motivate sustainable behaviors, not only by consumers but all participants in the Smart Grid. For example, electricity prices that truly reflect energy availability can influence consumers to shift their loads to minimize cost, and utilize distributed energy storage resources effectively [Joskow and Tirole, 2006]. Recognizing this opportunity, governments around the world have been deregulating their electricity markets to foster innovation. [Joskow, 2008]. However, some early failures such as the one that caused the California

energy crisis in 2000 demonstrate that transitions to competitive markets can be risky [Borenstein, 2002]. Consequently, there is a growing need to model and evaluate the expected dynamics of future electricity markets in a low-risk environment using software-based simulation.

We collaborated with several universities in the United States and Europe to develop *Power TAC*, an extensive distributed agent-based simulation environment for the retail Smart Grid [Ketter et al., 2013]. The focus of the simulation is on behavioral and economic aspects of agents in the future distribution grid, including their interactions with the transmission grid via wholesale markets. The Power TAC simulation contains realistic models of electricity consumers, producers, and markets, along with environmental factors, such as weather, that affect electricity production and consumption.

The diverse research teams that collaborate on Power TAC independently design and implement various agents in the simulation to mitigate the designer bias inherent in monolithic simulation environments. The techniques of *agent-based computational economics* [Tesfatsion, 2006] are used to study the impact of various assumptions about future tariff markets and also to develop economically-motivated strategies for the self-interested agents in the environment such as brokers and customers.

Power TAC is an example of a Trading Agent Competition (TAC) applied to electricity markets [Wellman et al., 2007].[1] Various market mechanisms and policy options can be applied to the simulation model and tested in open tournaments, where competition participants play the role of brokers while the competition infrastructure simulates the other agents shown in the tariff market scenario of Figure 1.1. Appendix D provides more information on the competition setting and the official game specification.

While the competition scenario motivates the development of profit-maximizing broker agents under the simulated policy regimes, Power TAC is also designed to serve as an offline research environment where other research goals can be pursued; *e.g.*, to study the impact of broader plugin electric vehicle (PEV) adoption, or the impact of rooftop solar deployments versus large centralized solar farms.

We anticipate that the Power TAC platform will enable numerous contributions to computational energy sustainability research. Moreover, the development of the platform has already presented interesting technical challenges—the remainder of this chapter addresses one such challenge where we develop generative hierarchical models to simulate the consumption and production capacities of Smart Grid customers.

[1]Previous TAC application domains have contributed invaluable insights to the computing and economic aspects of advertising auctions and supply chain management, for example. See http://www.tradingagents.org.

3.2 Bayesian Time Series Simulation

The scale of the scenario that can be modeled by a Smart Grid simulation environment is often a road block for efficient simulation. Power TAC aims to simulate a large variety of customer models, whose demand and supply depend on various factors such as installed load capacity, household size, geographic locale, day of week, month of year, temperature, humidity, cloud cover and wind speed. The simulations need to represent tens or hundreds of thousands of customers consuming or producing power under various tariffs offered by competitive broker agents in the market. Customers can vary along several dimensions, including the following:

- **ModelType** = $\{$*Individual, Population*$\}$
- **EntityType** = $\{$*Residential, Commercial, Industrial*$\}$
- **CapacityType** = $\{$*Consumption, Production, Storage*$\}$

3.2.1 *Problem*: Time Series Simulation

The need for truthful simulation mandates the use of fine-grained agent representations that model individual persons and/or appliances explicitly. [Gottwalt et al., 2011] describes a fine-grained customer model that simulates load profiles for homes equipped with smart appliances under real-time pricing (RTP) tariffs. [Guo et al., 2008] provide fine-grained models for household demand adaptation based on occupant comfort.

However, the computational requirements of such finely granular models make large-scale simulations difficult to run without very large amounts of computing hardware and sophisticated programming. It is highly beneficial to replicate the aggregate behavior of large groups of customers, *i.e.*, *populations*, using higher level statistical models while maintaining as much similarity as possible to the emergent behavior of corresponding fine-grained customer models.

Moreover, it would be beneficial to develop a reusable statistical framework that can be instantiated differently and fitted with different parameters to approximate the behavior of many heterogeneous customers without having to program each model from scratch. For example, we would like to model (i) an office building that only consumes power, (ii) a university campus that consumes power and also has some storage capacity, and (iii) a chemical plant that consumes power and also has some power production capability, using essentially the same model or code configured with different parameters.

Definition 3.1: A **bootstrap series** is a relatively short time series that is provided online during simulation to be used as a basis for forecasting to continue the simulation.

We assume that a sample of data from real world metering or fine-grained simulation is available *a priori* for training and an additional bootstrap series is made available online during simulation. We then need to generate more data, on behalf of a coarse-grained customer model agent, that simulates forward from the bootstrap data while borrowing characteristics from the training data.

Formally, let Y be the training time series representing the log-transformed production or consumption capacity of some Smart Grid customer model over N timeslots. Let $Z = Z_t$, where $t = 1..M$ and $M \ll N$, be a partial time series available to bootstrap our forecasting model. Let Z_t^* for $t > M$ be the remainder of the Z series to be used for evaluating forecasting accuracy. Let $D \in \mathbb{I}[1, 7]$ be the vector of length M identifying the day-of-week for each t and let $H \in \mathbb{I}[1, 24]$ be the vector identifying the hour-of-day.

We develop a generative framework that addresses the above requirements using a novel combination of hierarchical Bayesian [West and Harrison, 1997] [Berliner, 1995] and ARIMA [Cryer and Chan, 2008] methodology. We apply the framework to a specific sample data set from a fine-grained household customer model and study the process of fitting a model to that data. We also analyze the data generated by the resulting coarse-grained model to compare it with the sample data using several divergence metrics. Finally, we demonstrate some examples of how the coarse-grained model can then be reconfigured with subjective biases to generate alternate customer models.

3.2.2 Hierarchical Bayesian Model

We develop the methodology within the context of an example dataset generated using a fine-grained household customer model [Chrysopoulos and Symeonidis, 2009]. The model is based on real survey and metering data from the MeRegio project, an early Smart Grid deployment in Baden-Wurttemberg, Germany [Hirsch et al., 2010].

Figure 3.1 presents the available data; it consists of two time series, each representing the consumption or demand capacity of two villages. The fine-grained model represents each individual person or appliance as a discrete stochastic node and generates the aggregate series of Figure 3.1 by drawing from each of those nodes independently.

The time series for each village includes discrete demand (consumption capacity) measurements over 312 hours, *i.e.*, 13 days. Note the presence of two daily spikes in the morning and evening hours of each day in both series. Each discrete timeslot is also labeled with appropriate hour-of-day and day-of-week labels. These labels form two natural groupings for multilevel

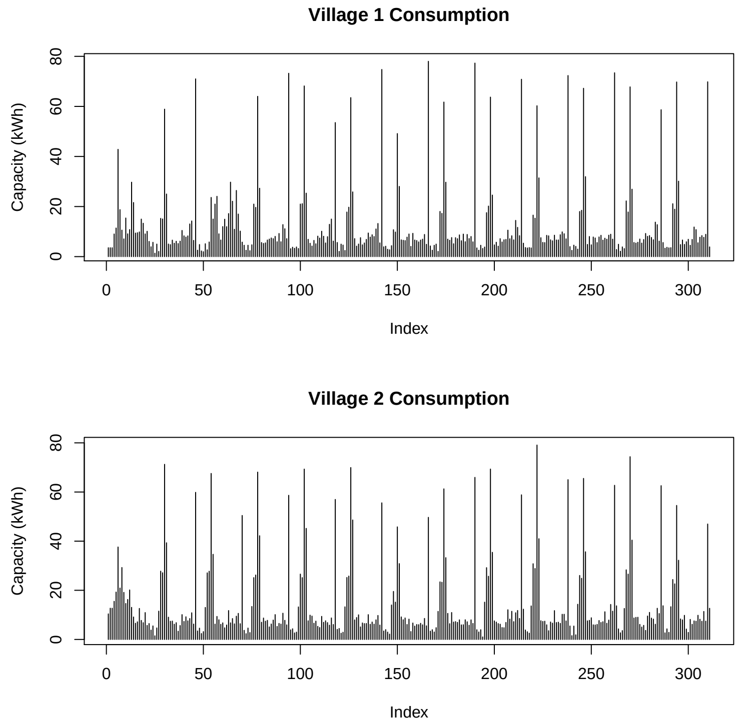


Figure 3.1: Consumption capacity of two small villages over 312 hours (13 days), a simulated by a fine-grained household customer model.

analysis: (i) hours labeled 1 to 24 with 13 samples per label, and (ii) days labeled 1 to 7 with only 1 or 2 samples per label. We explore these groupings further for random effects in our analysis.

3.2.2.1 ARIMA Methodology

A Box-Cox power transformation analysis suggests that the data be transformed using *log()* or *sqrt()* to achieve normality. We choose the *log()* transformation based on the Q-Q plots for each of the two transformations. Figure 3.2 summarizes the similarities and differences in the two log-transformed series using kernel densities and boxplots.

Furthermore, we evaluate the time series characteristics of the data in Figure 3.3. We see that the log-transformed series appears stationary with some seasonality. The ACF and PACF highlight significant coefficients for moving average (MA) components at lags of 1 and 24 and similarly the PACF highlights significant autoregression (AR) components also at lags 1 and 24. So we conclude that a $(1, 0, 1) \times (1, 0, 1)_{24}$ multiplicative seasonal model best fits the log-transformed data under the ARIMA methodology (see Appendix C).

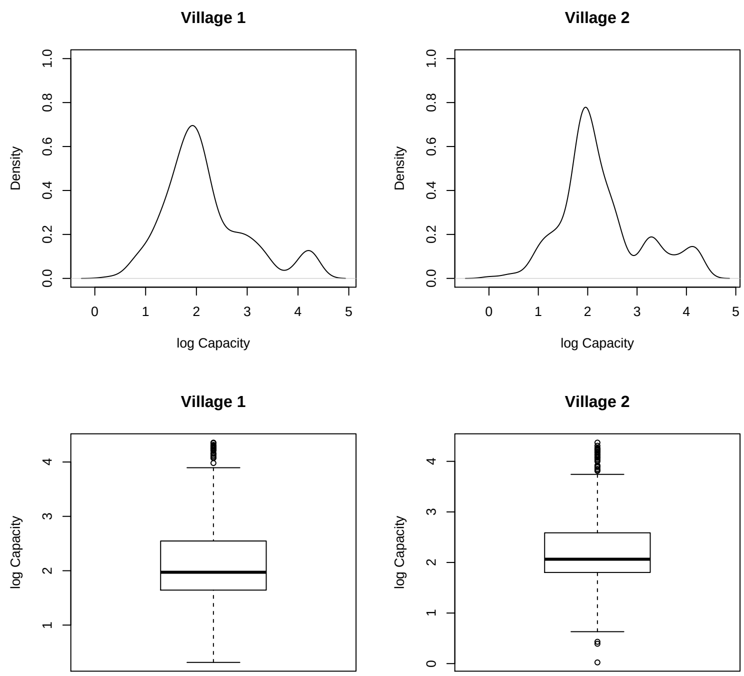


Figure 3.2: Kernel density plots and boxplots comparing the consumption capacity of the two villages.

We are assuming that we are to generate data for a simulation based on the full time series for village 1 and a small initial fragment of the time series for village 2. So, we will test the generated data against the remainder of the village 2 time series for forecasting accuracy. Note that there is no ground truth available online to periodically recalibrate the forecast since our goal here is to provide long range forecasting for simulation.

The estimated $(1, 0, 1) \times (1, 0, 1)_{24}$ multiplicative seasonal ARIMA model is of the form:

$$Y_t = Y_0 + \phi_1 Y_{t-1} + \Phi_1 Y_{t-24} + e_t + \theta_1 e_{t-1} + \Theta_1 e_{t-24} + \theta_1 \Theta_1 e_{t-25} \tag{3.1}$$

where Y_0 is the *grand mean*, the Y_t variables are the log capacity values at time t and the e_t values are the *innovations* at time t. The fitted coefficients for this model are:

	Y_0	ϕ_1	Φ_1	θ_1	Θ_1	$\theta_1\Theta_1$
SARIMA	2.1661	0.6162	-0.4473	0.9888	-0.7343	0.3285

When we use the fitted SARIMA model for forecasting, we encounter a characteristic problem with ARIMA forecasting whereby the autoregressive components of the fitted model deteriorate over time leading to forecasts that eventually revert to the grand mean of the series. This

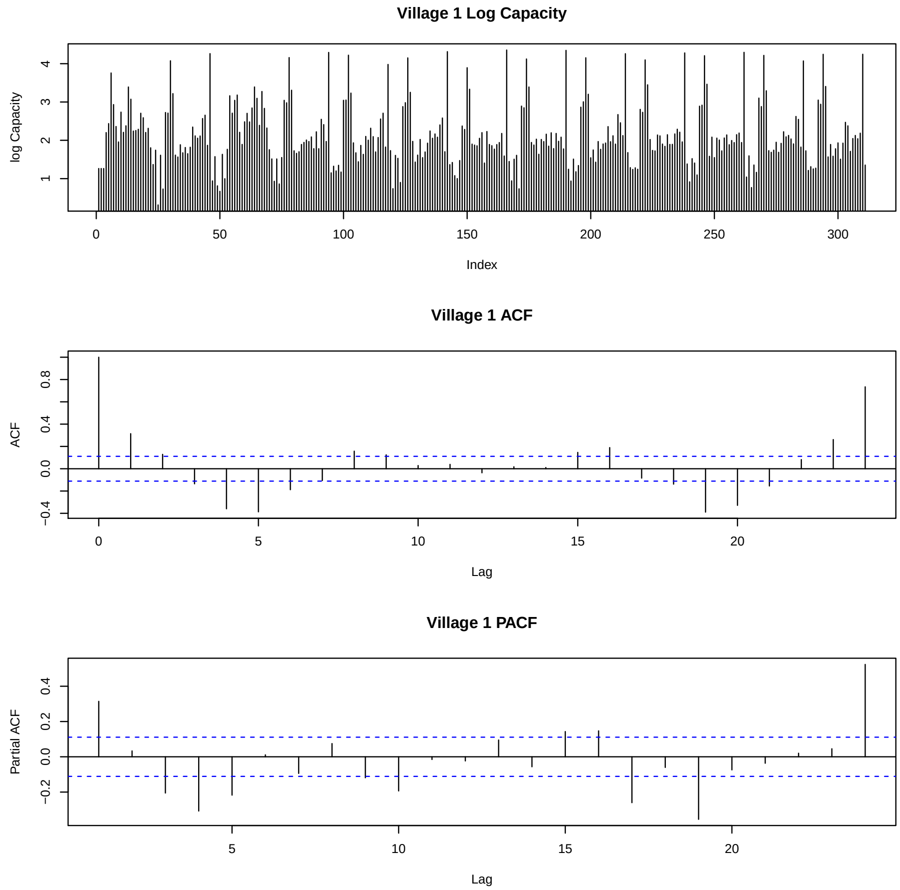


Figure 3.3: Time series characteristics of the consumption capacity of village 1 (the training series).

behavior is apparent in the top subfigure of Figure 3.4, which forecasts based on the full Y_t series, and more emphatic in the bottom subfigure, where the bootstrap time series, Z_t, is short and also doesn't capture the two-spiked daily cycle.

We would like to use daily spike information from Y_t as an informed prior in forecasting from Z_t but the mechanism for doing so is not obvious in ARIMA methodology. We could concatenate the two series but that can lead to inconsistencies at the concatenation point, so we do not pursue that approach.

3.2.2.2 Multilevel Regression

As an alternative, we use a grouped or multilevel regression model to fit coefficients to terms corresponding to those in the SARIMA model along with random effects for the day-of-week,

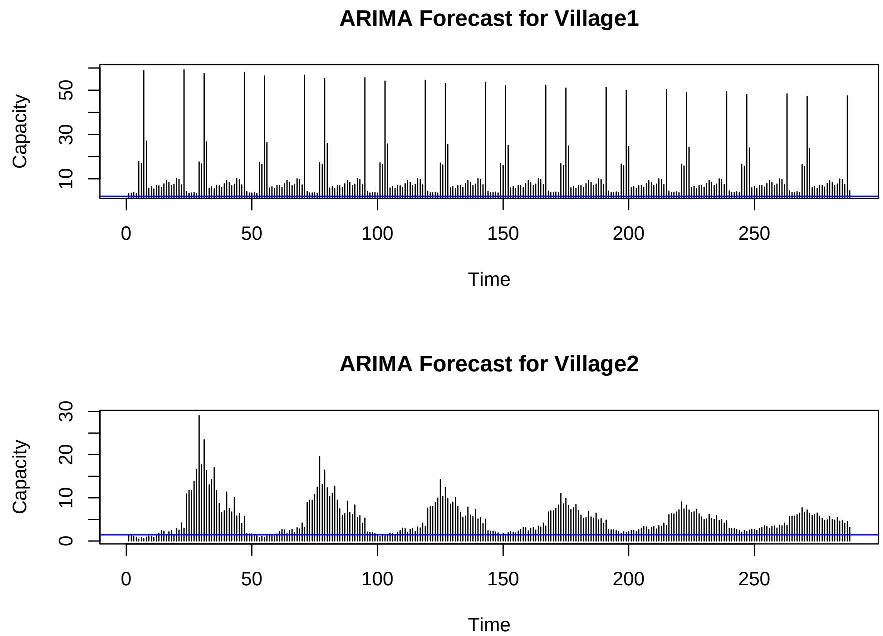


Figure 3.4: ARIMA model (Eq. 3.1) forecasts based on the full village 1 time series (top subfigure) and the first 24 hours of the village 2 series (bottom subfigure).

D, and hour-of-day, H.

$$Y_t \sim 1 + Y_{t-1} + Y_{t-24} + e_{t-1} + e_{t-24} + e_{t-25} + (1|D) + (1|H) \tag{3.2}$$

	1	Y_{t-1}	Y_{t-24}	e_{t-1}	e_{t-24}	e_{t-25}
LMER	1.4635	0.3390	−0.0180	−0.1303	−0.0405	0.0836

We find that the fixed effects coefficients, shown in the table above, vary significantly from the equivalent coefficients obtained using the fitted SARIMA model. We further find that only a few of the hour of day random effects are statistically significant and the day of week random effects are forced to zero. Not surprisingly then, forecasts of Z using the coefficients from the multilevel regression model worsen the problem of reversal to the mean compared to the SARIMA model.

3.2.2.3 Hierarchical Bayesian Methodology

The hierarchical Bayesian model [Gelman and Hill, 2007] we fit using Gibbs-sampling [Geman and Geman, 1984] is described below in Equations 3.3-3.19. Figure 3.5 represents the model using plate notation. Note that estimation starts at $t = 27$ to accommodate the lag of 26 time

steps needed by the the third moving average component in M_t. Also, this model forms only a subset of our overall model, completed in the next section.

$$Y_{1,t} \sim N(\widehat{Y}_{d,t}, \sigma^2);\ t = 27..N \tag{3.3}$$
$$Y_{2,t} \sim N(\widehat{Y}_{h,t}, \sigma^2);\ t = 27..N \tag{3.4}$$
$$\widehat{Y}_{d,t} \leftarrow Y_0 + Y_d[D[t]] + A_t + M_t \tag{3.5}$$
$$\widehat{Y}_{h,t} \leftarrow Y_0 + Y_h[H[t]] + A_t + M_t \tag{3.6}$$
$$A_t \leftarrow \phi_1 Y_{t-1} + \Phi_1 Y_{t-1} \tag{3.7}$$
$$M_t \leftarrow \theta_1(Y_{t-1} - Y_{t-2}) + \Theta_1(Y_{t-24} - Y_{t-25}) + \theta_1\Theta_1(Y_{t-25} - Y_{t-26}) \tag{3.8}$$

$$Y_0 \sim N(\widetilde{Y_0}, \sigma^2_{Y_0}) \tag{3.9}$$
$$\phi_1 \sim N(\widetilde{\phi_1}, \sigma^2_{\phi_1}) \tag{3.10}$$
$$\Phi_1 \sim N(\widetilde{\Phi_1}, \sigma^2_{\Phi_1}) \tag{3.11}$$
$$\theta_1 \sim N(\widetilde{\theta_1}, \sigma^2_{\theta_1}) \tag{3.12}$$
$$\Theta_1 \sim N(\widetilde{\Theta_1}, \sigma^2_{\Theta_1}) \tag{3.13}$$
$$\sigma \sim Unif(0, 100) \tag{3.14}$$

$$Y_d[d] \sim N(0, \tau^2);\ d = 1..7 \tag{3.15}$$
$$\tau \sim Unif(0, 100) \tag{3.16}$$

$$Y_h[h] \sim N(0, \eta^2);\ h = 1..24 \tag{3.17}$$
$$\eta \sim Unif(0, 100) \tag{3.18}$$

$$Y^*_t \sim N(\widehat{Y}_{1,t}, \sigma^2);\ t = 27..N \tag{3.19}$$

Before presenting the forecasting performance of this model, we briefly describe the steps taken to arrive at this model. We first fit coefficients, using Gibbs sampling, for the ARIMA features, *i.e.*, $Y_{t-1}, Y_{t-24}, e_{t-1}, e_{t-24}, e_{t-25}$ along with the grand mean Y_0. We then introduced the

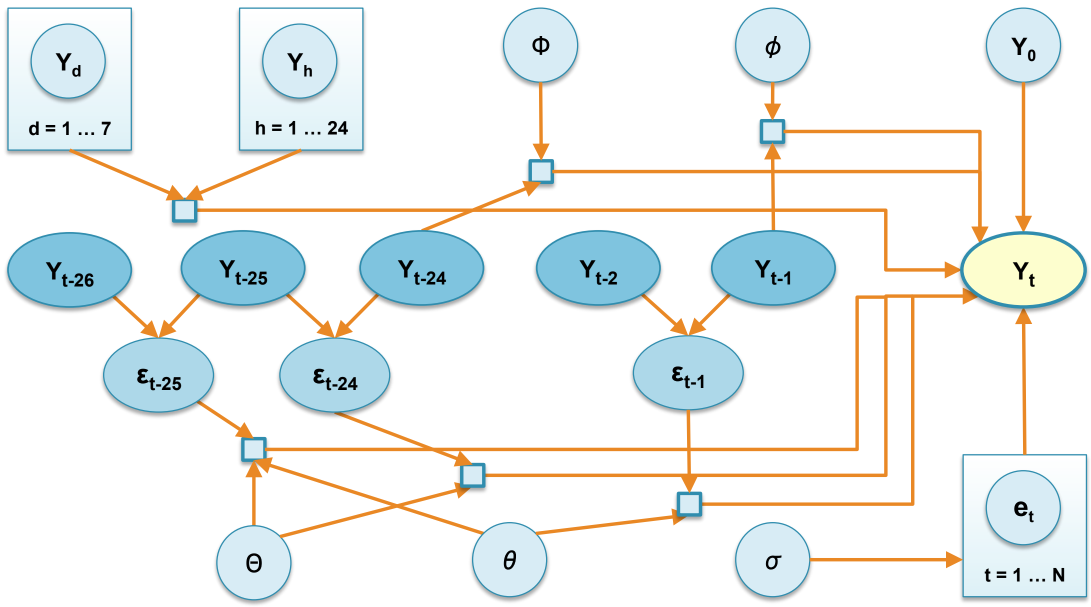


Figure 3.5: Graphical representation of the hierarchical Bayesian model in Eq. 3.3-3.18.

Y_d and Y_h terms using the following in place of Equations 3.5-3.6:

$$\widehat{Y}_{1,t} \leftarrow Y_0 + Y_d[D[t]] + Y_h[H[t]] + A_t + M_t \tag{3.20}$$

However, this approach does not work well because of the labeling ambiguity between the Y_d and Y_h terms, thus leading to very high standard deviations for the coefficients of both of those terms. So, instead we employ the mechanism of Equations 3.5-3.6, where we duplicate the original time series Y as Y_1 and Y_2. We then estimate coefficients for Y_d and Y_h separately using each of the two replicas. We acknowledge that we may be overestimating the coefficients for Y_d and Y_h by fitting them separately, but we address that concern further below when we describe how we utilize those coefficients.

Y_1, Y_2, D, H and N are provided as input to the model. Most of the remaining symbols in Equations 3.3-3.18 are self-describing based on earlier discussion, except the prior means and variances on the saved variables Y_0, ϕ_1, Φ_1, θ_1 and Θ_1. The prior means are set equal to the coefficients estimated by the original fitted SARIMA model. For the corresponding variances, we try two models:

1. the variances are also set equal to those estimated by the SARIMA Model, and

2. the variances are set to be very high thus creating vague priors.

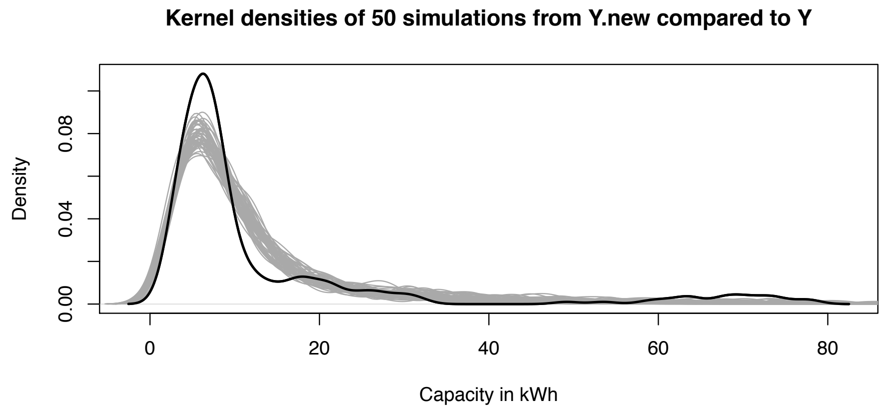


Figure 3.6: Kernel density plot of the true values of Y (dark line) compared to the plots for values simulated using Y^* (Eq. 3.19, light lines).

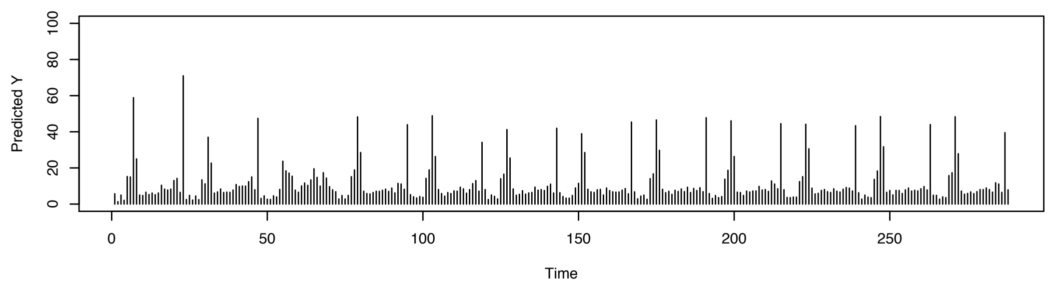


Figure 3.7: Long-range forecasts for village 2 consumption capacities using *true* histories values in the hierarchical Bayesian model in Eq. 3.3-3.18.

We compare the results of these two alternative variance models in the analysis below.

As a posterior predictive check, we use the Y_t^* simulated time series generated in Equation 3.19. We use kernel density estimates to see if the original series, Y_t, could have been generated by this model. The results in Figure 3.6 show that the density for Y is significantly higher in the 5-15 log capacity (in kWh) range than for Y^*. So, we conclude that the model of Equations 3.3-3.18 is not sufficient to meet our forecasting goals for long range simulation.

We also reach the same conclusion from the plots shown in Figures 3.7 and 3.8. We see in Figure 3.7 that the forecasted values for $\hat{Y}$ given the true historical values for Y are underestimates compared to the true values shown in the Village 1 series of Figure 3.1. Figure 3.8 shows

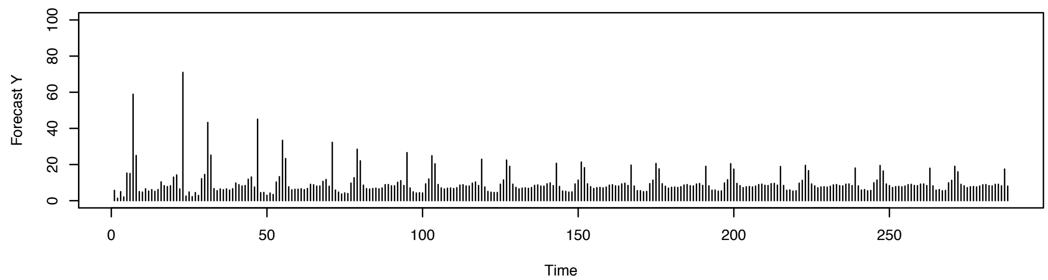


Figure 3.8: Long-range forecasts for village 2 consumption capacities using *simulated* historical values in the hierarchical Bayesian model in Eq. 3.3-3.18.

the forecast $Y_f = \hat{Y}$ derived by using the forecast Y_f values as historical values for $t > 26$ instead of the true Y values in Equations 3.7-3.8:

$$A_t \leftarrow \phi_1 \widehat{Y}_{t-1} + \Phi_1 \widehat{Y}_{t-1} \tag{3.21}$$

$$M_t \leftarrow \theta_1(\widehat{Y}_{t-1} - \widehat{Y}_{t-2}) + \Theta_1(\widehat{Y}_{t-24} - \widehat{Y}_{t-25}) + \theta_1\Theta_1(\widehat{Y}_{t-25} - \widehat{Y}_{t-26}) \tag{3.22}$$

The worse performance of Figure 3.8 compared to Figure 3.7 indicates that because we are not in an online setting where the forecast can be based on updated information, the multiplicative errors of the early forecast errors will continue to propagate for the rest of the simulation. If the forecasts are underestimates we see the long range forecast converge to the mean and, conversely, diverge if they are overestimates. So, instead of pursuing the elusive goal of the perfect forecast, we adopt an approach where we compensate for the multiplicative errors.

3.2.3 Augmented HBM Forecasting

Comparing the forecasts from the ARIMA methodology in Figure 3.4 and from the hierarchical Bayesian methodology in Figure 3.8, it may appear that we have not gained much, and lost significantly in computational complexity, by using the latter methodology. However, applying the hierarchical Bayesian has given us some tools that we can use to further enhance the model.

Specifically, we now have the fitted coefficients for Y_d and Y_h, which capture random effects or group intercepts for daily and hourly variations in the data. From visually examining Figure 3.8 and realizing that the fitted coefficients for the autoregressive components of the model are lower

than those in the ARIMA model, we hypothesize that *augmenting* the daily and hourly variations in the hierarchical Bayesian model will improve our forecast.

Furthermore, we note that the forecast deteriorates exponentially with time. So, we propose augmenting Y_f, our forecast from the hierarchical Bayesian model, by adding a logarithmic term that includes a weighted combination of the daily and hourly intercepts, Y_d and Y_h. The rationale for using a weighted combination takes into account our previously noted observation that Y_d and Y_h are both likely to be overestimates given that we estimated them separately, thus treating the group's intercepts as equal to zero.

Equation 3.23 summarizes the model described in detail in Equations 3.3-3.18. Equations 3.24-3.27 then describe how to extend the model to include the time-dependent logarithmic term which utilizes Y_d and Y_h.

$$Y_t^f \leftarrow Y_0 + Y_d[D[t]] + Y_h[H[t]] + A_t + M_t \tag{3.23}$$

$$Y_t^{bf} \leftarrow Y_t^f + \lambda \frac{log(t-26)}{log(N1-26)}((1-\gamma)Y_d[D[t]] + \gamma Y_h[H[t]]) \tag{3.24}$$

$$\lambda^*, \gamma^* \leftarrow \underset{\lambda,\gamma}{\operatorname{argmin}}\ KL_D(f_K(Y), f_K(Y^{bf})) \tag{3.25}$$

$$\lambda^*, \gamma^* \leftarrow \underset{\lambda,\gamma}{\operatorname{argmin}}\ \sum_t (Y_t^{bf} - Y_t)^2 \tag{3.26}$$

$$Z_t^{bf} \leftarrow Z_t^f + \lambda^* \frac{log(t-26)}{log(N2-26)}((1-\gamma^*)Y_d[D[t]] + \gamma^* Y_h[H[t]]) + N(0,\sigma^2) \tag{3.27}$$

In Equation 3.24, the weights of Y_d and Y_h in the exponential combination are determined by a parameter γ. The combination is then multiplied with a normalized time factor, $log(t-26)/log(N1-26)$, where $N1$ is the number of time steps in Y, further multiplied by a scaling factor λ.

We then define choosing the right values for λ and γ as an optimization problem that minimizes the *distance* between Y and the augmented model forecast Y^{bf}. The distance need not be symmetric, so we can use an asymmetric measure such as KL-divergence or a symmetric distance such as a point-wise sum of squares over the discrete time steps in the forecast. In order to be able to use KL-divergence, we need probabilities to compare, so we obtain kernel density estimations, f_k, for Y and Y^{bf} and use those in the comparison. Equations 3.25-3.26 represent these two options.

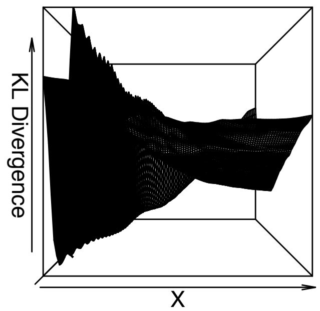


Figure 3.9: Surfaces representing the *distance* between the true village 1 series and its simulation forecast using KL-divergence (left), and sum of least squares (right).

In our experiments, we use a brute-force search over combinations of λ and γ:

$$\lambda \leftarrow i = 1...100;\ i \in \mathcal{I} \tag{3.28}$$

$$\gamma \leftarrow 0.01j;\ j = 1...100;\ j \in \mathcal{I} \tag{3.29}$$

Figure 3.9 shows the two surfaces representing the divergence metric computed by the two methods. We observe that the surface on the right corresponding to the sum of least squares method is smoother and therefore a candidate for a more efficient optimization method such as gradient descent or the Gauss-Newton method. In our experiments, both methods take approximately the same amount of computational time and yield nearly the same values for the optimal parameters, λ^* and γ^*. However, we anticipate that other training samples for Y may yield results with greater difference and therefore present both options here. It is also possible to combine the estimates from both options to compute the values for λ^* and γ^*.

The optimized values of λ^* and γ^* are then used in Equation 3.27 to generate an augmented forecast for the online time series Z. Equation 3.27 replicates the structure of Equation 3.23 with $N2$ representing the forecast horizon or length of Z^*. It also adds Gaussian noise with variance σ^2 where σ is a parameter fitted by the hierarchical Bayesian model in Equations 3.3-3.4.

In Figure 3.10, we analyze the fit of the forecast Y^{bf} using kernel density plots as we did in Figure 3.6. We now see that the original series Y appears to fit well with the simulations for the forecast model for Y^{bf}. Figure 3.11 shows how the augmentation step improves our forecasting performance by eliminating the reversal to the mean that we observed with previous models.

Figure 3.12 shows the difference between the forecast from the training series Y and the forecast from the bootstrap portion of the test series Z. We see that the two forecasts disagree more in

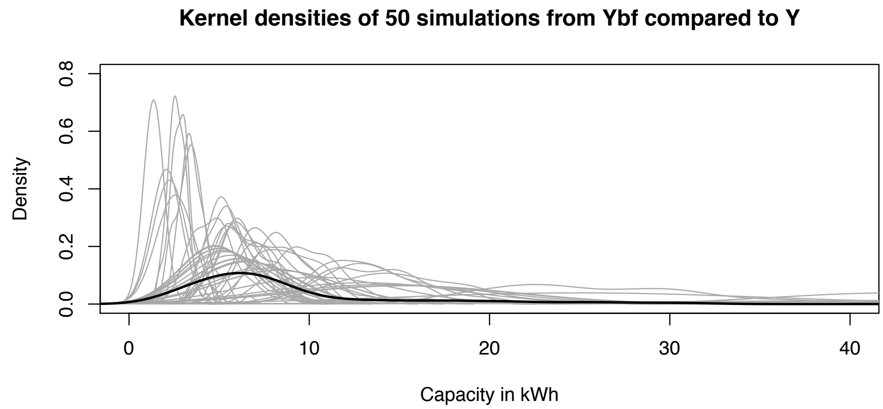


Figure 3.10: Kernel density plot of the true values of Y (dark line) compared to the plots for values simulated using Y^{bf} (Eq. 3.24, light lines).

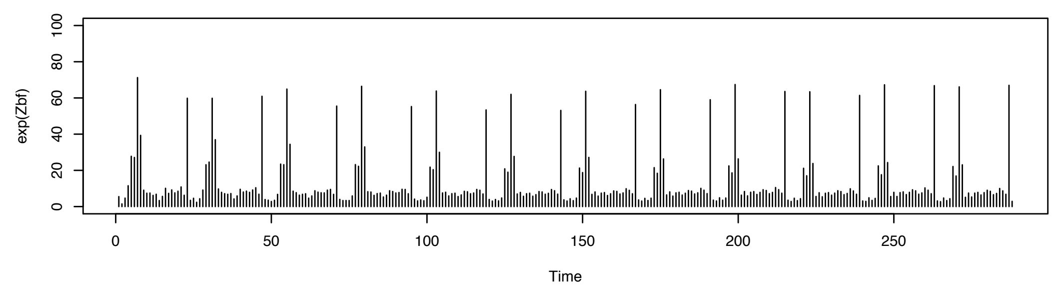


Figure 3.11: Long-range forecasts for the village 2 consumption capacities using *simulated* historical values in the *augmented* hierarchical Bayesian model of Eq. 3.23-3.27.

earlier time steps where Z's bootstrap sample exerts greater influence while the difference disappears over time. This demonstrates that we meet our simulation goal whereby we use information from Y as a prior but also take into account the available portion of Z in making a forecast for Z.

Figure 3.13 shows the performance of the various forecasting methods discussed thus far using a different metric. The x-axis plots *tolerance*, a threshold for the percentage error beyond which the forecast consumption capacity for a future timeslot is considered wrong. For example, at the 20% tolerance level, given that the true capacity at some timeslot t' is $Z^*_{t'}$, then a forecast that equals $1.2\, Z^*_{t'}$ would be classified as 1 whereas $1.21\, Z^*_{t'}$ would be classified as 0. The y-axis measures *accuracy* which is the percentage of timeslots over the forecast horizon which are classified as 1.

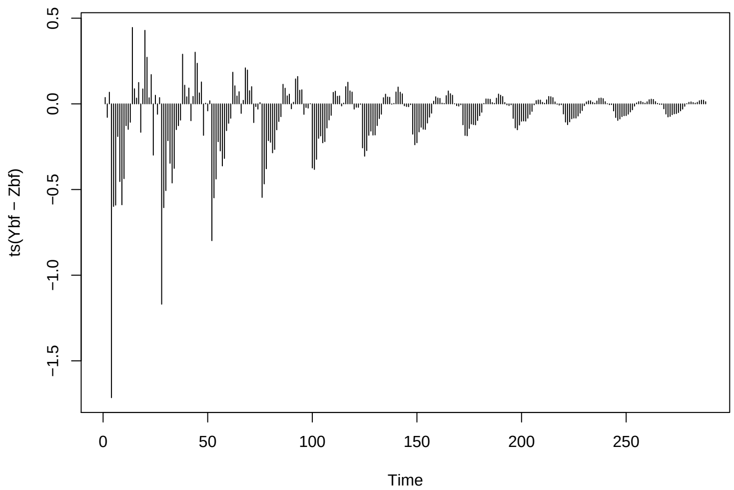


Figure 3.12: Difference in consumption capacities forecasted from the village 1 training series Y and the village 2 bootstrap series Z.

The accuracy of the forecast for the online test series Z using conventional ARIMA methodology (solid black line) serves as the benchmark. The accuracy for the training forecast, Y^f, using the variances from the fitted seasonal ARIMA model as the variances for the priors in the hierarchical Bayesian model of Equations 3.9-3.13 (dashed red line) improves upon benchmark. The equivalent plot for Y^f with those variances set at 10^4 thus providing very little information in the priors is shown by the dotted blue line; we see that the ARIMA priors perform slightly better at lower tolerance levels and the vague priors perform better at higher tolerance levels but the difference is marginal. The forecast accuracy for Y^{bf} using the augmented HBM model of Equations 3.23-3.27 (dash-dotted magenta line) universally performs better than the three previous methods. Finally, the accuracy for the testing forecast Z^{bf} using the augmented HBM model (dashed green line) performs slightly worse than the corresponding training forecast Y^{bf} as is to be expected, but performs universally better than the other three methods with the largest differences in the critical 10-40% tolerance range.

3.2.3.1 Simulation using Subjective Biases

A significant benefit of using the hierarchical Bayesian methodology for our framework is that we can impose additional subjective biases on the expected forecast for a given time series. We can provide these biases as priors on the daily and hourly weights, Y_d and Y_h, for example.

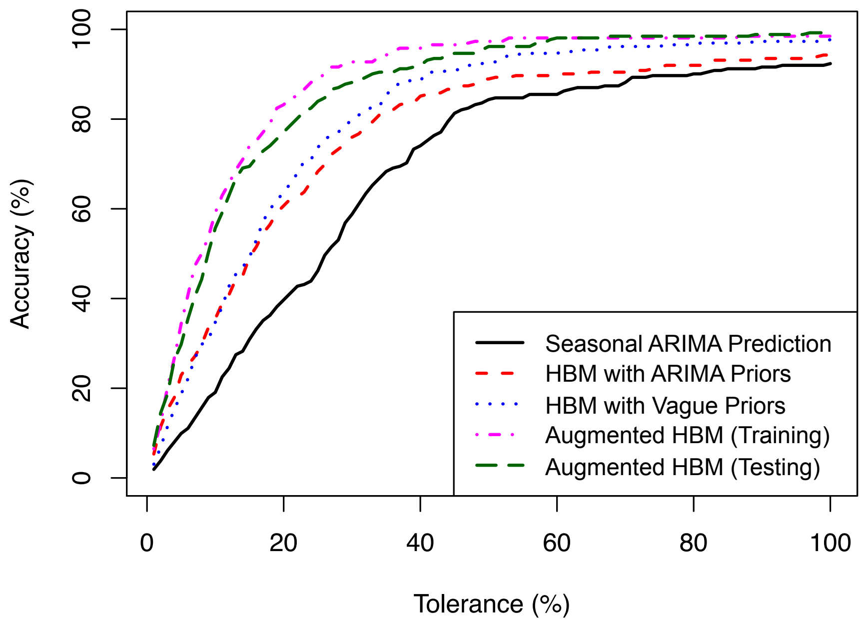


Figure 3.13: Accuracy of various forecasting models measured against a range of error-tolerance levels.

Alternately, we can impose them *a posteriori* as shown in Equation 3.30 where the subjective forecast, Z^{sf}, skews the augmented forecast, Z^{bf}, using hour-, day- and month-specific weights relative to the capacity at the start, t_0, of the time series.

$$Z_t^{sf} \leftarrow Z_t^{bf}\ \omega_H(t)\ \omega_D(t)\ \omega_M(t) \tag{3.30}$$

These weights may be computed taking into account additional factors not described in our model such as the typical occupancy profile of households in a region or daily temperature variations. Figure 3.14 illustrates a concrete example where the top subfigure shows a static set of *prior* hourly weights based primarily on the occupancy profiles of households in the region represented by the model. The subfigure on the bottom shows a modified set of weights that are based on the prior weights and also the daily variations of temperature in the region. The hourly temperature values can be expected values for the time of the year or dynamic values forecasted for the next hour by a real-time weather service. Similarly, we can provide daily weights that forecast higher than average capacity on Saturdays but lower than average on Sundays, for example. Figure 3.15 shows the output of such a skewed forecast in the middle subfigure; the top subfigure shows the unskewed model forecast.

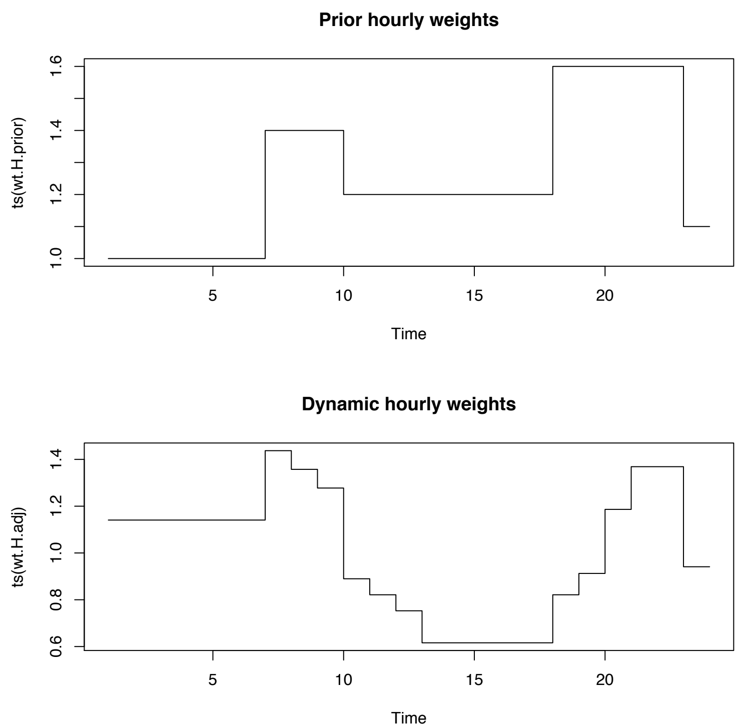


Figure 3.14: Example of subjective hourly weights used as priors (top) to derive posterior weights that also account for temperature (bottom).

Another example concerns exogenous factors that we do not expect to be captured by the training series, Y. In our scenario, the consumption capacity of a set of households is likely dependent on the *tariff rates* or electricity prices that they are subject to. In other words, we may expect that if tariff rates are lower, the households will consume more electricity on average. We can reflect such an assumption by modeling the *elasticity* [Pindyck and Rubinfeld, 2004] of consumption capacity to various electricity rates, V, relative to a benchmark rate, V^*, as shown in Equation 3.31.

$$Z_t^{ef} \leftarrow Z_t^{sf} + elasticity(t,\, V_t,\, V^*) \tag{3.31}$$

Note that the elasticity factor may be negative, for example if electricity rates become less favorable. The bottom subfigure of Figure 3.15 shows the simulated series layered model of Equation 3.31 where consumption is lower in the second half of the series because of higher

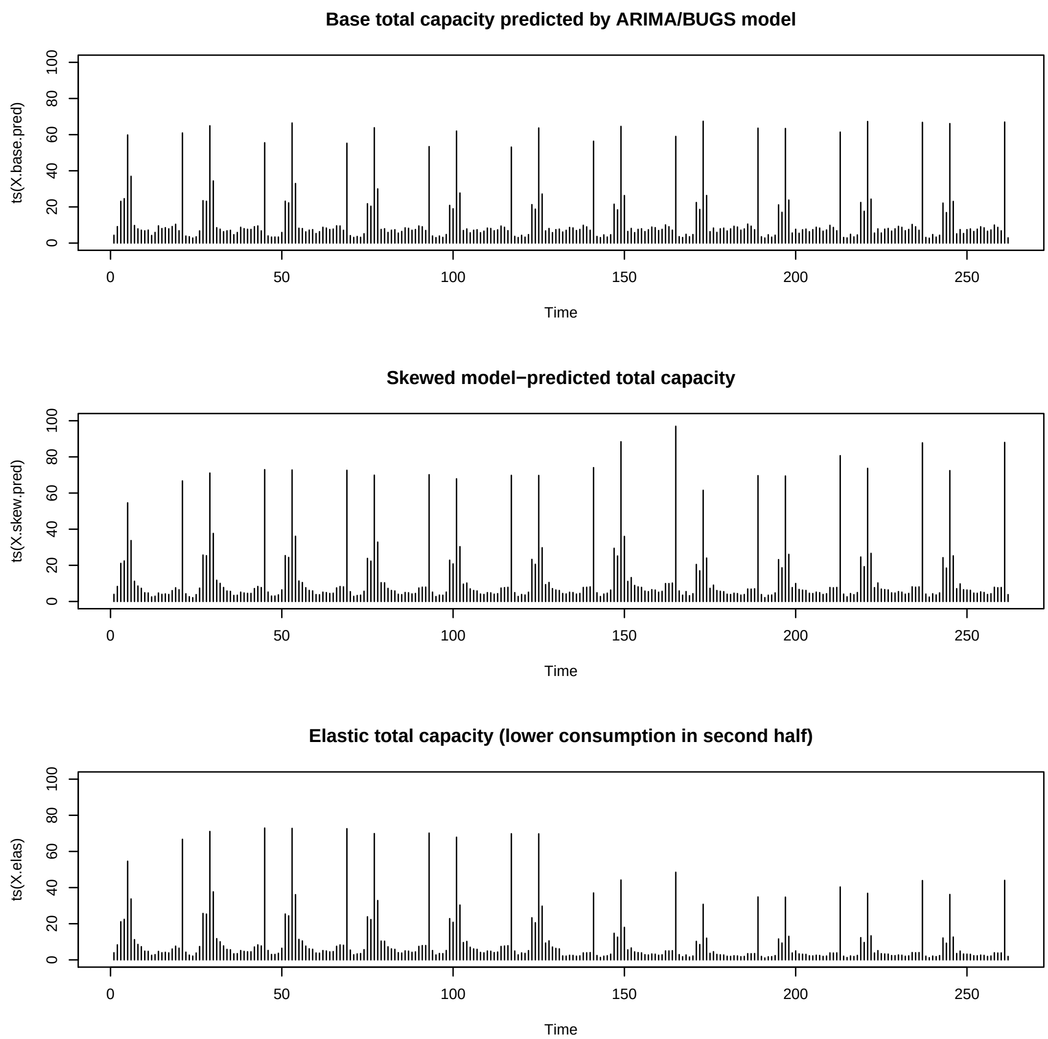


Figure 3.15: Example of exogenous factors being used to inform the priors of the augmented hierarchical Bayesian model.

tariff rates. Over time, we would then endogenize these additional factors into the Bayesian model and apply the associated biases *a priori* so that they can be modulated by the data.

3.3 Chapter Summary

In this chapter, we highlighted the importance of software-based simulation in the evaluation of new market structures and agent behaviors for future Smart Grid markets. Power TAC offers a distributed agent-based simulation environment to facilitate such evaluation. We encountered the problem of simulating a long range time series using a combination of offline and online data. This problem is challenging because there is no access to ground truth data that can be revealed over time. This constraint implies that the forecasts must themselves be used as historical values, which leads to a multiplicative effect on any forecasting errors that worsens with the length of the simulation. While we analyze and address this problem within the context of Smart Grid customer agent simulation, similar situations arise in other simulation domains, where our augmented hierarchical Bayesian methodology may also be applicable.

We used data from a fine-grained household consumption model to learn how a coarse-grained model can simulate a time series that approximately replicates the essential characteristics of the given data. We emphasize that our focus in this work is not to recover the true parameters that were used to generate the fine-grained data and neither is our focus on trying to model parameters representing specific individual person or appliance behaviors in the population. We instead aim to:

(i) model generically applicable factors that can represent a broad set of customers,

(ii) determine appropriate hierarchical models based on those factors, and

(iii) fit the coefficients or distribution assumptions for those factors based on specific data that we are trying to replicate.

Our augmented hierarchical Bayesian methodology for long range forecasting achieves these goals. In the next chapter, we explore how these factors can be combined in a versatile framework to simulate many customer agent types and drive their behaviors.

Chapter 4

Adaptive Customer Agents

A key challenge in the Power TAC platform is the simulation of the vastly heterogenous behaviors of customers of various sizes (*e.g.*, residential, commercial, industrial) and functions (*e.g.*, consumer, producer, hybrid). So, we developed a *factored customer model* framework to represent the multiscale decision-making responsibilities of a diverse set of customer types. We use a generalized set of *factors*, most of them represented as probability distributions, to define the intrinsic *tariff selection* and *capacity management* behaviors of various customer types and their responses to stimuli from the simulation environment. Additionally, the instantiated customer models can be nested to represent customers at arbitrary granularity. This chapter first describes this framework and some example instantiations in Section 4.1.

Then in Section 4.2, we delve into our first customer agent strategy, a decision-theoretic approach using stochastic optimization. We tackle the scenario of multi-dwelling consumers, such as apartment buildings and rural electricity cooperatives, where each dwelling maintains autonomy over its consumption behavior but is cooperative with the other dwellings to achieve shared benefits such as lower cost of electricity. We assume the existence of a centralized *utility optimizer*, which models its interaction with each dwelling as a semi-cooperative relationship. The stochastic *quantal response* equilibrium computed by the optimizer shows that the dwellings can autonomously maximize their own self interest and yet achieve the shared benefits of lower aggregate demand volatility and corresponding cost savings.

4.1 Factored Customer Models

The emergent behavior observed in simulations is key to understanding the scenarios that need to be considered in developing Smart Grid technology. The granularity at which simulations are

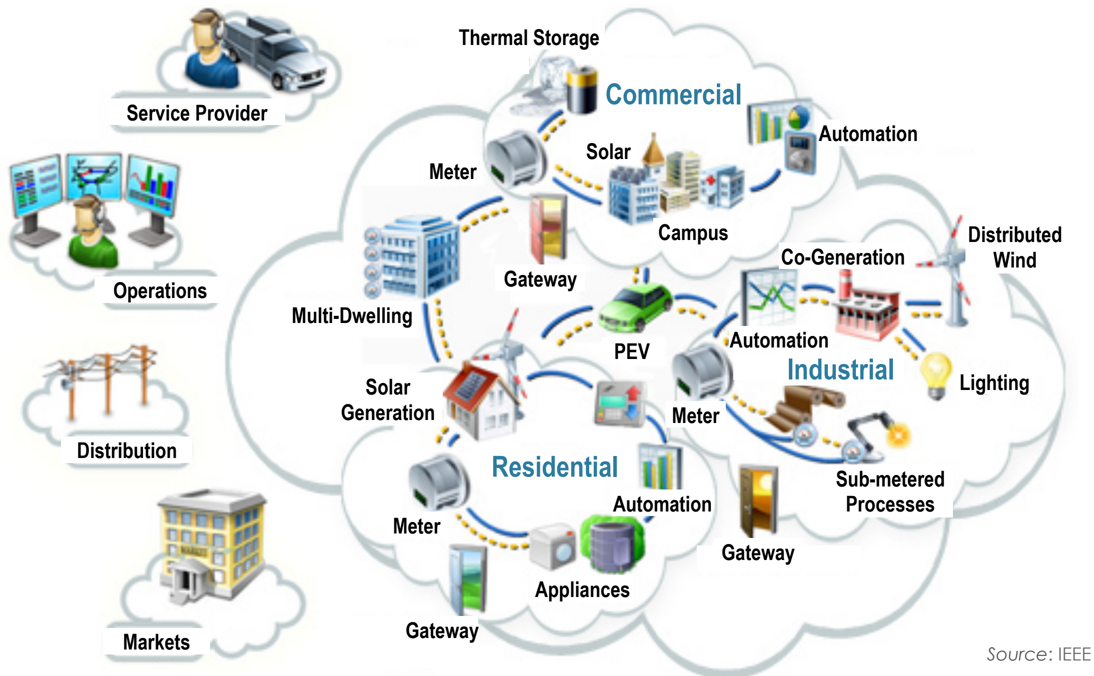


Figure 4.1: An illustration of a typical combination of heterogeneous entities involved in the distribution Smart Grid that need to be modeled in simulation.

conducted and the realism with which the components of the simulations are represented directly influence the types of lessons that can be learned. For example, simulations of the nation-wide grid may be necessary to stress test long-haul transmission line capacities whereas simulation of a single *smart home* may yield lessons on appliance load coordination. Customers can be modeled simply as consumers, producers and storage facilities or more explicitly as suburban homes, office complexes, solar farms, electric vehicles and so on. Along another dimension, residential customers can be represented at the granularity of a single household with each appliance modeled separately, at the aggregate behavior of the entire household, or over collections of many households. The Institute of Electrical and Electronics Engineers (IEEE) illustrates the many customer types and related entities as shown in Figure 4.1. The effort required to model this diversity of options is a critical challenge.

In following sections, we present a versatile agent-based *factored customer model* that enables rich simulation scenarios across distinct customer types and varying agent granularity by leveraging a generic customer representation that can be suitably parameterized.[1] We formulate the decisions to be made by each Smart Grid customer as a multiscale decision-making problem.

[1]Our use of the term *factored* is analogous to its use in factored MDPs, factor graphs in probabilistic models, and factored (*i.e.*, multiplicative) ARIMA models. We intend for it to convey that the model's behavior is characterized by a composition of its determining factors.

We introduce a *utility optimizer* as a component that manages the multiscale decisions and briefly describe how it may be embodied in the real world. We also contribute an algorithm for adaptive capacity management using decision-theoretic approximation of multiattribute utility functions over multiple agents.

4.1.1 Customers in Tariff Markets

We assume that the modeled customers operate in competitive tariff markets where they have a choice of brokers and possibly multiple tariffs from each of them. Customers respond to tariff price changes [Gottwalt et al., 2011] and have a range of preferences over tariff terms. For example, some are willing to subscribe to variable rate tariffs if they have the opportunity to save by adjusting their power usage, while others are willing to pay higher prices for the simplicity of fixed rate or simple time-of-use (TOU) tariffs.

Moreover, customers with controllable capacities can participate in *demand-side management* (DSM) initiatives and brokers can offer special tariffs for them. Separate tariffs may be offered for charging electric vehicles, which could limit charging during high-demand periods, or even offer to pay the customer for feeding electricity back into the grid at certain times. Tariff contracts may include both usage-based and per-day charges, fixed and varying prices for both consumption and production of electricity, rates that apply only above a specified usage threshold, signup bonuses, and early-withdrawal penalties.

4.1.2 Multiscale Decision-Making

We observe that Smart Grid customers are faced with a *multiscale decision-making* problem along at least the following two dimensions, as illustrated in Figure 4.2:

1. **Temporal**: Customers must simultaneously manage their current capacity levels given their tariff prices and also their tariff choices given their expected capacity levels. While capacity management occurs at high frequency, tariff selection occurs at a lower frequency.

2. **Contextual**: For example, a single household must consider the optimal behavior of each appliance individually but also of all appliances together and similarly of the household unit versus its *neighboring* households. While this contextual dimension is loosely related to the spatial dimension, it can also be applied more broadly using abstract definitions of *neighborhood*.

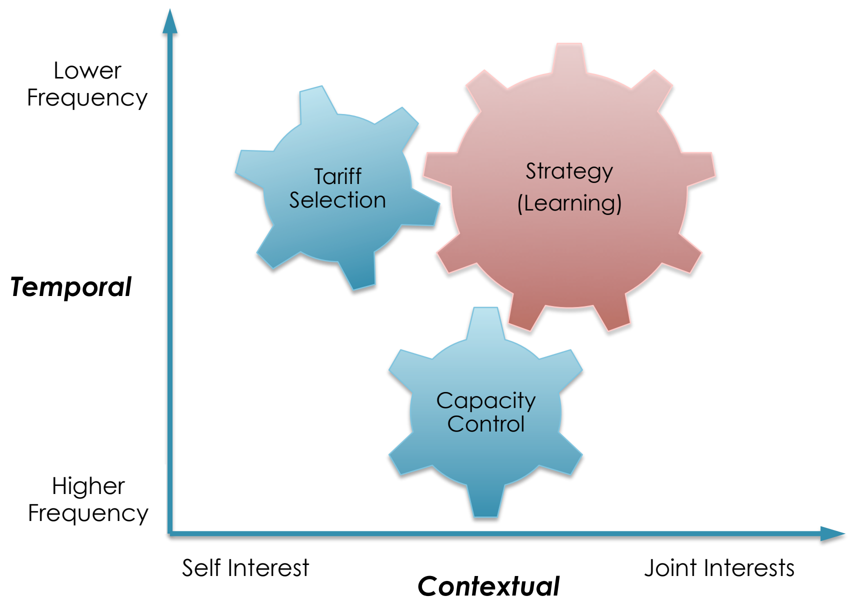


Figure 4.2: Smart Grid customer agents are faced with decision-making tasks that are interrelated along *temporal* and *contextual* dimensions.

Therefore, at each time step, t, a customer must perform the following optimization:

$$\operatorname*{argmax}_{y_t} \mathrm{U}_S(p_t, y_t, \mathrm{U}_N(y_t)) \tag{4.1}$$

where y_t is the capacity level, U_S is a self-utility function, p_t is the applicable tariff price, and U_N is the neighborhood-utility function. Then at certain less frequent time steps, t', that occur every τ time steps, the following optimization is needed:

$$\operatorname*{argmax}_{z \in \mathcal{Z}_{t'}} \mathrm{U'}_S(\vec{p}_{\{z,t'\}}, Y_{t'}, \mathrm{U'}_N(Y_{t'})) \tag{4.2}$$

where $\mathcal{Z}_{t'}$ is the set of applicable tariffs available at t', $Y_{t'}$ is the expected *capacity profile* over the horizon τ, $\vec{p}_{\{z,t'\}}$ is the vector of expected prices under a tariff z over τ at t', and the utility functions, $\mathrm{U'}_S$ and $\mathrm{U'}_N$, are evaluated over τ.

Integrating these decisions into one multiscale problem facilitates learning and application of strategies to coordinate the decisions along both dimensions [Barber, 2007]. [Wernz and Deshmukh, 2010a] formalize the concept of multiscale decision-making in the domain of organizational behavior using an analytical game-theoretic approach. We use this formulation to develop adaptive and learning strategies for customer agents later in this chapter and also in Chapter 5.

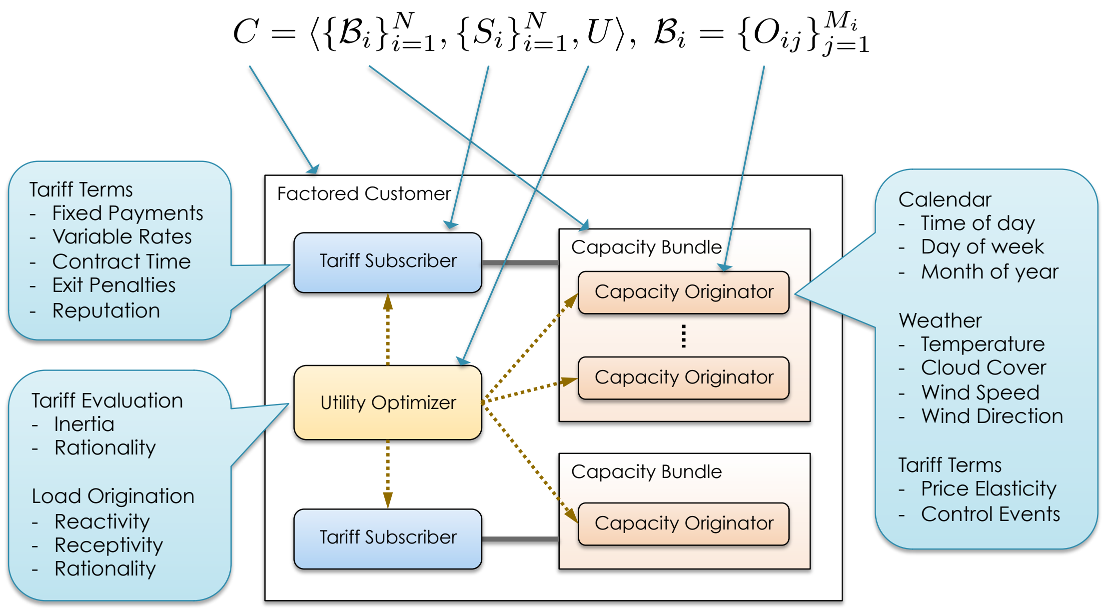


Figure 4.3: Example *factored customer* modeled with 3 capacity originators in 2 capacity bundles—some of the *factors* used to control the model's instantiated behavior are also shown.

4.1.3 Factored Customer Representation

Let a Smart Grid customer, C, be defined as:

$$C = \langle \{\mathcal{B}_i\}_{i=1}^N, \{S_i\}_{i=1}^N, U \rangle,\ \mathcal{B}_i = \{O_{ij}\}_{j=1}^{M_i}$$

$\mathcal{B}_i$ is a *capacity bundle*, which contains one or more *capacity originators* O_{ij}. S_i is a *tariff subscriber* and U is a *utility optimizer*. Figure 4.3 illustrates this composition; the 1-to-1 correspondence between $\mathcal{B}_i$ and S_i is shown with solid lines while the dotted arrows indicate recommendations from U.

4.1.3.1 Capacity Originator

Definition 4.1: A **capacity originator** represents a unit of power consumption or production whose behavior is is driven by its *base capacity generator* and several *influence factors*.

Definition 4.2: The **base capacity generator** in a capacity originator is either an arbitrary probability distribution or a model-based *time series generator*. The time series generator may use the augmented hierarchical Bayesian simulation methodology described in Section 3.2 or some other time series forecasting technique.

The capacity originator generates an original capacity level, y_t^o, for each discrete time step, t, by drawing from the base generator distribution or by obtaining the next prediction from the time series generator.

The original capacity level, y_t^o, is then adjusted according to some subset of the following influence factors:

- **Calendar**: The factors, time-of-day $\in \mathbb{I}[1, 24]$, day-of-week $\in \mathbb{I}[1, 7]$, and month-of-year $\in \mathbb{I}[1, 12]$, are given weights, which are multiplied with y_t^o such that if the weight is less than 1, then the resulting capacity is decreased and vice versa.

- **Pricing**: The multiplicative weight of this factor is computed based on absolute tariff prices (measured in currency units) or a price elasticity function applied to deviation of current prices from a configured *benchmark* price.

- **Weather**: Multiplicative weights, again based on segmented real values or elasticity functions applied to benchmark deviations are computed for the following factors: temperature, wind speed, wind direction, and cloud cover.

The adjusted capacity level, y_t^a, obtained as the product of y_t^o and each factor's multiplicative weight is then used to forecast capacity profiles. These capacity profiles may be further modified using adaptive capacity management techniques like the one presented in Section 4.2. Important additional influence factors that enable adaptive capacity control are also described in that section. A capacity originator can be viewed variably as an appliance, or recursively as one or more customers. It can also be an autonomous control agent or a decision-support interface to humans who manually control capacity.

4.1.3.2 Capacity Bundle

Definition 4.3: A **capacity bundle** is an aggregation of capacity originators with the constraint that all originators in the bundle must be of the same *capacity type*. The capacity type can be categorized coarsely as {*consumption, production, storage*} or more finely with types such as *household consumption*, *wind production*, and *electric vehicle storage*.

Typically, one bundle is assigned to a single tariff, however when the bundle represents a collection of grid-connected entities as in a farming cooperative, the bundle can allocate segments of its population to different tariffs.

4.1.3.3 Tariff Subscriber

Definition 4.4: A **tariff subscriber** is an autonomous or human agent that manages the assignment of a capacity bundle to one or more of the available tariff choices. The agent is modeled using a multinomial logit choice model where the utility of each tariff choice is assumed to be given.

Two additional factors determine the outcome of the tariff selection process:

- **Inertia**: This is modeled as a probability distribution, a draw from which decides whether the subscriber maintains its corresponding capacity bundle in its current tariff subscriptions or whether it considers reassignment.

- **Rationality**: This factor, $\lambda \in [0, 1]$, determines the degree to which the tariff utility values, $\mathrm{U}_{\tau}^{S}(z)$, associated with the tariff choices influence tariff selection:

$$Pr(z) = \frac{e^{\lambda \mathrm{U}_{\tau}^{S}(z)}}{\sum_{\mathcal{Z}} e^{\lambda \mathrm{U}_{\tau}^{S}(z)}} \tag{4.3}$$

  The probability values thus derived can be used for (i) random selection of a single tariff for the entirety of the associated capacity bundle, usually where a model represents a single grid-connected entity, or (ii) for proportional allocation of the capacity bundle, where bundles can be partitioned, to multiple tariffs, *e.g.*, a population of household customers [Ketter et al., 2011].

4.1.3.4 Utility Optimizer

In their survey of customer behavior under real-time pricing (RTP) tariffs offered by over 20 utilities in the United States, [Barbose et al., 2005] observe limited elasticity to price changes. They note that this may be due to inadequate customer-side automation that can capitalize on opportunities arising from real-time price changes by actively managing consumption or production capacities. The utility optimizer component of our factored customer model forms an optionally deployed intelligent autonomous agent to serve this goal. We extend its scope to automate the frequent tariff subscription decisions that have become more important for customers in recent years with increasing competition in retail markets. We describe its optionally included *adaptive capacity management* algorithm in detail in Section 4.2.

We conclude this section with some examples of how our factored model can be instantiated to represent varying customer types and agent granularities.

1. *Fine-grained Household Model*: Individual appliances are represented as separate capacity originators with each originator drawing its base capacity from an appropriate probability distribution.

2. *Coarse-grained Household Model*: All consumption appliances are represented in aggregate as one capacity originator that draws from a time series generator, and rooftop solar panels form another originator in a separate bundle.

3. *Research University Model*: General use loads such as lighting and HVAC in campus buildings are collected in one consumption bundle, ICT equipment and research labs are collected in a different consumption bundle, a gas-fired local generation facility is modeled as a separate originator in its own production bundle.

4. *Farming Cooperative Model*: All consumption for all farms is modeled as one originator in its own bundle, and the windmills on each farm are modeled as separate originators but collected in one wind production bundle.

In each of the above examples the tariff subscriber and utility optimizer may be autonomous agents integrated into the capacity control automation or be separate services that help inform humans and execute their commands.

Figure 4.4 illustrates a rich set of correlated, yet distinct, consumption (positive capacity) and production (negative capacity) patterns that are generated for a Power TAC competition setting. Each pattern depicts a population model of some set of customers, represented as a factored customer model instantiation.

4.2 Stochastic Capacity Optimization

Demand side management (DSM) has been an important focus area for Smart Grid research over the past decade [Strbac, 2008]. Smart Grid customers are steadily acquiring distributed renewable generation capabilities; the promises and challenges of this evolution have increased the urgency of progress in DSM-related research [Gomes, 2009] [Amin and Wollenberg, 2005]. However, achieving active participation from customers in the management of their electricity demand and supply is a complex problem with numerous scenarios that are difficult to test in field projects, *e.g.*, [NREL, 2012].

Prominent among the approaches being studied to achieve active customer participation in DSM is one based on offering customers financial incentives through variable rate tariffs. Some studies have shown that *adaptive capacity management* by customers in response to such tariffs

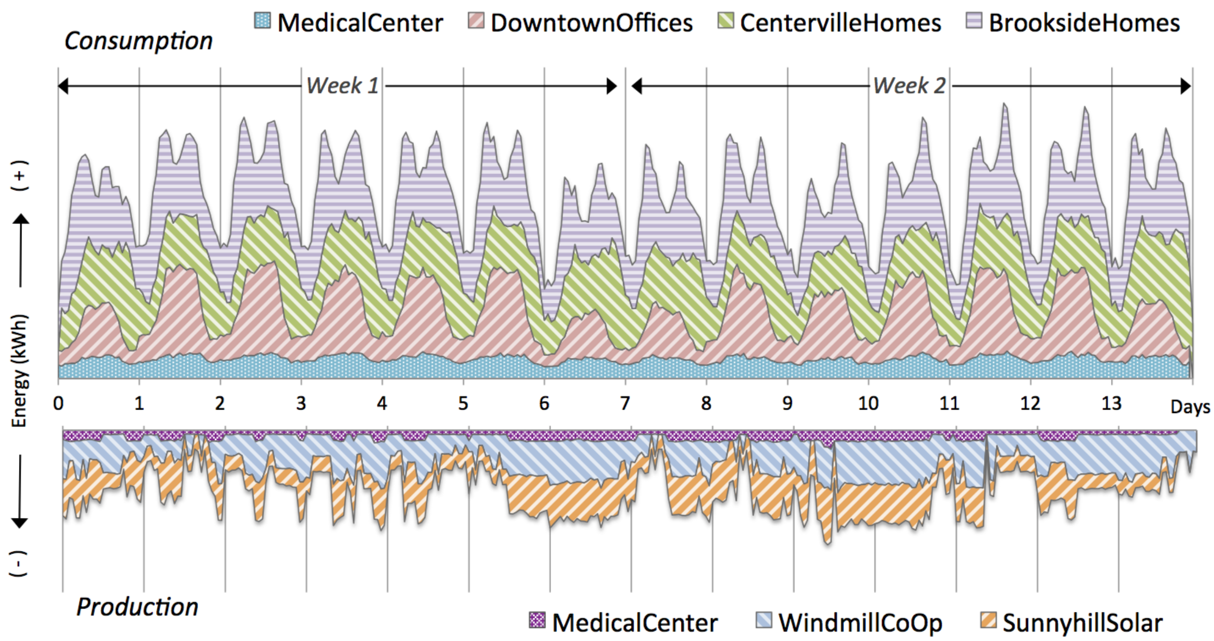


Figure 4.4: Sample of diverse consumption (+ve) and production (-ve) capacities generated by factored customer model instances representing various customer populations.

leads to detrimental peak-shifting behavior. Our stochastic optimization algorithm contributes an effective solution to this *customer herding* problem.

4.2.1 *Problem*: Adaptive Capacity Management

[Vytelingum et al., 2011] and [Voice et al., 2011] describe the problem of shifting peaks under real-time pricing (RTP) amongst micro-storage agents such as plugin electric vehicles (PEV). In this problem, many agents independently converge their loads on short time intervals with lower expected prices thus leading to undesirable load peaks. For example, if prices are high until 8pm, many PEV owners will start charging their cars soon after 8pm causing a shifted peak at that time. [Gottwalt et al., 2011] observe the same phenomenon in their simulations and call it the *avalanche* effect.

[Ramchurn et al., 2011] illustrate this effect using Figure 4.5 and refer to it as the *herding* problem – terminology which we adopt – and also propose an adaptive algorithm that imposes inertia on proportions of customers, so that some customers are denied the opportunity to optimize their capacity profiles at certain times, to mitigate the impact of this effect. [Voice et al., 2011] propose a mechanism based on penalties for deviation from past behavior to achieve a similar result. While artificially imposing inertia or penalties is a reasonable approach from the

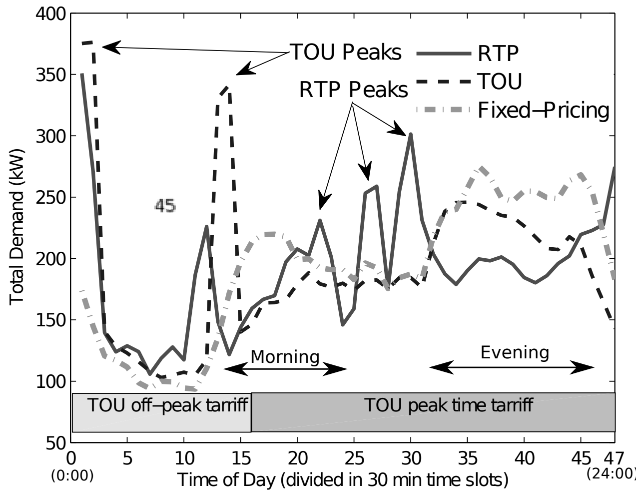


Figure 4.5: Example of undesirable shifted peaks in demand (*i.e.*, consumption capacities) because of consumer responses to TOU and RTP tariffs. [Ramchurn et al., 2011]

perspective of mechanism design, we present an alternate solution. Our stochastic optimization algorithm mitigates the herding problem using behavioral agent modeling [McKelvey and Palfrey, 1995] multiattribute utility theory [Wellman, 1985] and decision-theoretic agent behavior [Horvitz et al., 1988].

Concretely, consider the scenario of a rural electric cooperative where the *members* – typically farms, households, and small businesses – are not serviced by a distribution utility but instead have to maintain their own distribution infrastructure and buy electricity as a group. Consequently, they share the cost of the infrastructure and the consumed electricity. Depending on the size of the cooperative, they may buy electricity directly from wholesale markets or subscribe to tariffs offered by brokers. Without loss of generality, given our focus on capacity management, we assume that they subscribe to tariffs.

The intrinsic sharing of costs induces a shared or *neighborhood utility*. Moreover, each member preserves autonomy over their own consumption levels and patterns, *i.e.*, capacity profiles, so that they have their own *self utility* distinct from their neighbors. The combination of self-interest and joint interests creates a semi-cooperative scenario.

Specifically, we assume independent cooperative members who are:

- *not* willing to give up control over their capacity profiles,

- willing and able to share forecasts of their capacity profile, and

- willing and able to take *profile recommendations*, defined shortly, for capacity shifting.

In Section 4.1.2, we defined the multiscale decision-making problem faced by Smart Grid customers as two utility maximization problems with different temporal and contextual components. In the next section, we present how we define those utility functions in the context of this rural cooperative example and present an approximate algorithm to maximize them, which also addresses the herding problem.

4.2.2 ϵ-Quantal Response Equilibrium

Definition 4.5: A **capacity profile**, ρ_H, is a time series of capacity values up to the horizon, H.

We define the distance between two capacity profiles as the sum of squared point deviations:

$$D(\rho_H, \tilde{\rho}_H) = \sum_{t=1:H} (\rho_t - \tilde{\rho}_t)^2 \tag{4.4}$$

4.2.2.1 Profile Permutations

Definition 4.6: An admissible **profile permutation**, $\tilde{\rho}_H$, of a given profile ρ_H (i) has the same cumulative capacity over H, and (ii) has a minimum consumption capacity no smaller than ρ_H or a maximum production capacity no greater than ρ_H.

Without loss of generality, we assume ρ_H is a consumption capacity and therefore its permutation, $\tilde{\rho}_H$, satisfies the following constraints:

(i) $\sum_t \tilde{\rho}_H(t) = \sum_t \rho_H(t)$ over H, and

(ii) $\min(\tilde{\rho}_H) \geq \min(\rho_H)$ for consumption capacity.

We enlist two methods for generating admissible capacity profile permutations towards the goal of adaptive capacity management via *capacity shifting*:

- **Temporal Shifts**: A lag-ϑ permutation, $\rho_{H,\vartheta}$, is obtained by rotating the capacity elements of ρ_H by ϑ time steps. $\mathcal{R}_T^{\rho_H}$ is then the size H set of all temporal shifts for ρ_H. Intuitively, this procedure is applicable when the duty cycle of a long-running capacity originator, such as a pool pump, can be deferred for ϑ time steps.

- **Balancing Shifts**: Let ρ_j be the permutation obtained from ρ_i by setting $\rho_j = \rho_i$, computing $x = 0.5\,range(\rho_i)$, subtracting x from $\max(\rho_H)$ and adding x to $\min(\rho_H)$. ρ_k can then be similarly obtained from ρ_j and so on. $\mathcal{R}_B^{\rho_H}$ is then the set of balancing shifts obtained

by recursively computing ρ_j and adding it to $\mathcal{R}_B^{\rho_H}$ until $range(\rho_j) < \epsilon$ or until the size of $\mathcal{R}_B^{\rho_H}$ reaches a threshold.

This procedure generates permutations that are *flatter* than the previous profile at each iteration. It is applicable when the capacity bundle consists of mostly independent short-running loads that can be individually shifted, *e.g.*, run the clothes dryer at a different time, or when a capacity originator can operate at various capacity levels, *e.g.*, run the air conditioner at 10am so that it is not needed at 12pm.

These two methods for generating profile permutations are generic and do not take into account the specific appliances or other contributors to the forecast capacity profile of each member, *i.e.*, whether a spike at 2pm is caused by an electric car charging or by the clothes dryer. It would certainly be possible to create permutations based on such specific information, if it is available natively or can be computed by disaggregating the aggregate profile [Kolter et al., 2010]. Such specific permutations will be more accurate and feasible than the generic permutations we adopt here, but do not materially alter our algorithm below, which can be applied equally well with specific permutations.

4.2.2.2 Adaptive Capacity Originator

To represent each member of the cooperative, we extend the capacity originator definition in the factored customer representation to define an *adaptive capacity originator*, which can receive a *profile recommendation* from the utility optimizer and adapt, *i.e.*, shift, its capacity profile according to the recommendation.

Definition 4.7: A **profile recommendation** is an ordered map of permutations of the current forecast, $\hat{\rho}_H$. Each entry in the recommendation maps a permutation to a real-valued *permutation score*, $\tilde{\rho} \rightarrow \upsilon \in \mathbb{R}$.

We describe the computation of scores below, but for now, we declare that a higher score is preferable to the adaptive capacity originator. The recommendation map is then ordered by reverse-sorting the included permutations using the score values.

The self-utility of a permutation is defined as:

$$\underbrace{\mathrm{U}_S(\tilde{\rho}_H)}_{\text{self utility}} = \underbrace{\Delta f_p(\tilde{\rho}_H)}_{\text{cost savings}} + w_D \underbrace{D(\tilde{\rho}_H, \hat{\rho}_H)}_{\text{shifting disutility}} + w_N \underbrace{\mathrm{U}_N(\tilde{\rho}_H)}_{\text{shared utility}} \tag{4.5}$$

Δf_p is a function that computes the change in expected payment relative to the forecast profile and U_N is a neighborhood utility function described below. The distance from $\hat{\rho}_H$ to $\tilde{\rho}_H$ represents

the *shifting disutility* to the capacity originator. Thus, the utility of each permutation is a weighted combination of the change in payment, the shifting disutility and the neighborhood utility.

A permutation $\tilde{\rho}$ is only included in a recommendation if its associated expected payment (debit if consumption, credit if production) is better than that of the forecast profile, $\hat{\rho}$. This ensures that members of the cooperative do not end up paying more after adaptation than they would have otherwise. However, a member may incur higher shifting disutility by adopting a recommended permutation.

When an adaptive capacity originator receives a recommendation from a utility optimizer, it filters the map of permutations for feasibility. If the utility optimizer uses the generic *temporal* and *balancing* permutation-generating methods, it is expected that more of the recommended permutations will be discarded compared to when specific permutations are used.

We model the originator's decision-making process for selecting amongst the feasible permutations, using three responsiveness factors—*reactivity*, *receptivity*, and *rationality*. These factors are intended to capture the possibility that the capacity originator may in fact be modeling a human decision-maker.

Definition 4.8: The **reactivity** of a capacity originator is the probability that it will at least consider shifting to a recommended profile permutation.

Definition 4.9: The **receptivity** of a capacity originator is the probability that it will adopt the permutation with the highest score amongst the feasible permutations. This implies that it follows the advice of the utility optimizer whenever possible.

Definition 4.10: The **rationality** of a capacity originator is a factor in its probabilistic choice over the set of feasible permutations in a profile recommendation. For example, as in our experiments, this may be the λ factor in a multinomial logit choice model similar to the one used for tariff selection in Equation 4.3, with the score values of each permutation, $\upsilon_{\tilde{\rho}}$, provided by the utility optimizer used as the *self utility* values computed by Equation 4.1.

The overall procedure that the originator applies to choose an adapted capacity profile $\rho(t)$ to execute at time t is summarized in Algorithm 4.1.

4.2.2.3 Stochastic Utility Optimizer

We have already assumed that the adaptive capacity originator representing each member of the rural cooperative is willing and able to (i) share its forecast capacity profiles, and (ii) act on recommended permutations. We now consider two scenarios in the design of the algorithm for the utility optimizer of the factored customer model representing the cooperative:

Algorithm 4.1 **ADAPTIVE-CAPACITY-ORIGINATOR**$(t, \vec{L}, \mathrm{U}_s)$

$\vec{L}' \leftarrow$ **FeasibilityFilter**$(\vec{L})$
$x \sim$ *Random*$(0, 1)$
if $x < x_{\text{reactivity}}$ **then**
    $\rho[t] \leftarrow \hat{\rho}[t]$
else
    if $x < x_{\text{reactivity}} + x_{\text{receptivity}}$ **then**
        $\rho[t] \leftarrow \vec{L}'[0]$
    else
        $\rho[t] \leftarrow$ **ProbabilisticChoice**$(\vec{L}', \mathrm{U}_s, x_{\text{rationality}})$
    end if
end if

Scenario 1: Assume additionally that the self utility function, U_s of Equation 4.1 as instantiated by Equation 4.5, of each capacity originator is known to the utility optimizer. Assume also that the $x_{\text{reactivity}}$, $x_{\text{receptivity}}$, and $x_{\text{rationality}}$ parameters used by each originator in Algorithm 4.1 are known to the optimizer.

Scenario 2: Make no additional assumptions.

These two scenarios represent the extremes of a cooperation spectrum between the capacity originators and the utility optimizer. Note that the utility optimizer is fundamentally assumed to be cooperative, therefore an adversarial relationship does not form one extreme of the cooperation spectrum. Both extremes are semi-cooperative—even in scenario 1, the capacity originators are semi-cooperative with each other because their self-interested utility is not identical to their joint utility and they are semi-cooperative with the utility optimizer because they are not giving up control of capacity shifting.

Even with the additional assumptions of scenario 1, the utility optimizer cannot simply perform a brute-force optimization over all the capacity originators and their profile permutations because the adaptive capacity originator's decision procedure, shown in Algorithm 4.1, is stochastic due to the probability draws related to reactivity and receptivity. Moreover, if the originator chooses to consider all feasible permutations, even if its $x_{\text{rationality}}$ is known, its permutation choice may be subject to a further random draw in its probabilistic choice model. Therefore, we do not attempt to design a deterministic capacity optimization algorithm.

Instead, our *stochastic utility optimizer* adopts an approach based on Monte Carlo sampling to derive approximately optimal profile recommendations for each originator. The approach is outlined in Algorithm 4.2. The algorithm is invoked at time t with the set $\mathcal{O}$ of capacity origi-

Algorithm 4.2 **UTILITY-OPTIMIZER**$(t, \mathcal{O}, \mathcal{Y}, \mathcal{P}, \mathcal{M}, T)$

for o in $\mathcal{O}$ **do**
  $(y_o, \rho_o, M_o) \leftarrow (\mathcal{Y}[o], \mathcal{P}[o], \mathcal{M}[o])$
  UpdateOriginatorModel(M_o, y_o)
  $\vec{\tilde{\rho}}_o \leftarrow$ **GenerateProfilePermutations**(t, ρ_o)
end for
for i in $1..$MaxIterations **do**
  for o in $\mathcal{O}$ **do**
    $\dot{\rho}_o \leftarrow$ **DrawProfilePermutations**$(\vec{\tilde{\rho}}_o, M_o)$
  end for
  $u_N[i] \leftarrow \sum_{o \in \mathcal{O}}$ **ComputeNeighborhoodUtility**$(M_o, \dot{\rho}_o, T)$
  for o in $\mathcal{O}$ **do**
    for $\tilde{\rho}$ in $\vec{\tilde{\rho}}_o$ **do**
      $\Delta u_N[\tilde{\rho}] \leftarrow$ **ComputeNeighborhoodUtility**$(M_o, \tilde{\rho}, T) - u_N[i]$
      $u_s[\tilde{\rho}] \leftarrow$ **EstimateSelfUtility**$(M_o, \tilde{\rho}, \Delta u_N[\tilde{\rho}], T)$
      $v_o[\tilde{\rho}] \leftarrow$ **ExponentialUpdate**$(u_s[\tilde{\rho}])$
    end for
  end for
end for
for o in $\mathcal{O}$ **do**
  $\vec{L} \leftarrow$ ReverseSort$(\vec{v}_o)$
  CommunicateRecommendation$(o, \vec{L})$
end for

nators, a set $\mathcal{Y}$ of realized capacity values at $t-1$ per originator, a vector $\mathcal{P}$ of forecast capacity profiles as of t per originator, a set $\mathcal{M}$ of *originator models*, and the current tariff T.

Each originator model in $\mathcal{M}$ represents the utility optimizer's beliefs about the corresponding originator's (i) self-utility function, U_s, (ii) $x_{\text{reactivity}}$ and $x_{\text{receptivity}}$ parameters, and (iii) probabilistic choice model including $x_{\text{rationality}}$ if applicable. This encapsulation maintains flexibility in the algorithm so that it can be applied for various levels of the cooperation spectrum between scenario 1 and 2 described above. With sufficient history, the weights for shifting disutility, w_D, and neighborhood utility, w_N, in U_s may be learned using regression. Similarly, if $x_{\text{reactivity}}$ and $x_{\text{receptivity}}$ are unknown, they can be modeled as *Beta*(α, β) distributions that can be initialized to the uniform distribution *Beta*(1,1) and updated at each time step using binomial observations. Such updates to the model are abstractly represented by the *UpdateOriginatorModel* function in the algorithm. At each time step, the algorithm then generates admissible permutations using the temporal and balancing shift mechanisms or more specific heuristics if available.

Then the algorithm repeats the following procedure up to a maximum number of iterations. One profile permutation is drawn for each capacity originator o using the probabilistic choice

model in its corresponding model, M_o. Unless a more informed model is specified, we assume a multinomial logit distribution—this is also known as *logit quantal response*. A neighborhood utility value is then computed as a function of all drawn permutations. The neighborhood utility function, U_N, can be designed to encourage different metrics such as minimum variance of aggregate capacities across all originators, minimum aggregate shifting disutility over all originators, and so on. Next, the algorithm iterates through all the capacity originators and invokes the following sub-procedure for each originator: (i) holding constant the permutations for all other originators, iterate through the permutations of the chosen originator and compute the change in neighborhood utility values, Δu_N, for each of them, (ii) exponentially update the *score*, υ, for each permutation using the Δu_N values in the originator's assumed self-utility function, U_s. After the maximum iterations are reached, a *recommendation*, $\vec{L}$, for each originator is created as a map of permutations ordered by decreasing score and communicated to the originator.

The adaptive capacity management approach described in this section avoids the herding problem typically seen with shifting under variable rate tariffs. This is at least partly due to the logit *quantal response* model we employ, *i.e.*, the assumption of a multinomial logit probabilistic choice model over the recommended permutations. The logit quantal response model ensures that shifted capacities are assigned equitably to equivalent future time steps instead of being greedily shifted to the next time step with low expected prices. Furthermore, the weighted self-utility function, Equation 4.5, which explicitly accounts for neighborhood utility, also helps prevent many capacity originators converging on the same time step.

In traditional game theory, the equilibrium that emerges from all players adopting quantal response strategies is not surprisingly called the *quantal response equilibrium* [McKelvey and Palfrey, 1995]. Since our stochastic utility optimizer approximately computes this equilibrium on behalf of all the capacity originators, we refer to the emergent outcome of the utility optimizer's algorithm as ϵ-*quantal response equilibrium*.

As we demonstrate in experimental results, the adapted capacity profiles are less volatile than the raw profiles, thus making it easier for energy brokers and physical service providers to anticipate them, which in turn leads to greater social welfare. The customers themselves reap financial rewards from this surplus because the permutations they consider adapting to are constrained to have better expected payment values than their originally forecast profiles.

4.2.2.4 Experimental Results

The development of the factored customer model representation described in this chapter was driven by the needs of the Power TAC simulation environment and research goals. We success-

fully deployed the model framework, with a rich set of instantiations, in a number of Power TAC tournaments since 2011. The capacity patterns for one particular set of instantiations is illustrated in Figure 4.4. The XML file used to generate these instantiations is shown in Appendix F.

Note that the instantiated models include consumption-only, production-only and hybrid models. They also include commercial and industrial models in addition to residential models. The diversity of these instantiated models, all configured using a uniform set of factors specified through the XML configuration, demonstrates that our model achieves our stated goal of versatility. The flexibility of the factor model enables research using the Power TAC platform for a variety of future Smart Grid scenarios with heterogenous customer population models.[2]

Figure 4.6 illustrates the behavior of the multinomial logit tariff selection model of Equation 4.3 for a representative population model of 30,000 residential customers. Each shaded area represents the percentage of customers allocated to a particular tariff. At the start of simulation, all customers are allocated to the default tariff offered by the distribution utility. As competitive brokers enter the market, they publish new tariffs, some of which are deemed attractive enough by the factored customer model to allocate some percentage of the population to those tariffs, while others fail to attract any customers. The brokers iterate over their tariff terms to try and acquire additional customers, so more tariffs are introduced over time. Presumably, these tariffs are designed with sufficient knowledge of existing tariffs in the market such that most of them manage to attract at least some customers when they are introduced. Over time, the customer population is split across a subset of tariffs, such that the proportion of the population of each tariff is determined by the assumed rationality, λ, of the population.[3]

In another representative experiment, we modeled 10,000 residential customers as profiled in the real world data from the MeRegio study in Germany [Hirsch et al., 2010]. We instantiated 100 capacity originators each representing 100 customers. The capacity patterns of each originator were modeled using the long range time series simulation model of Section 3.2 and trained on the output of the fine-grained household simulation model of [Gottwalt et al., 2011]. We then simulated brokers who offered TOU tariffs with higher prices in the afternoon and evening as is typical in many real-world TOU tariffs.

We observed that when the adaptive capacity management was enabled, the customers typically attained 5-12% cost savings. Perhaps more importantly, the resulting shifted capacities do not demonstrate *herding* and have significantly lower volatility. Figure 4.7a shows the aggregated original capacities of the population (dashed black line), the shifted capacities with only temporal

[2] Note however, that the model framework software is sufficiently generic and encapsulated such that it could be ported to other Smart Grid simulation environments for wider applicability.

[3] The ontology that governs the structure of the offered tariffs is presented in Appendix E along with some real world example tariffs in deregulated US states such as Pennsylvania.

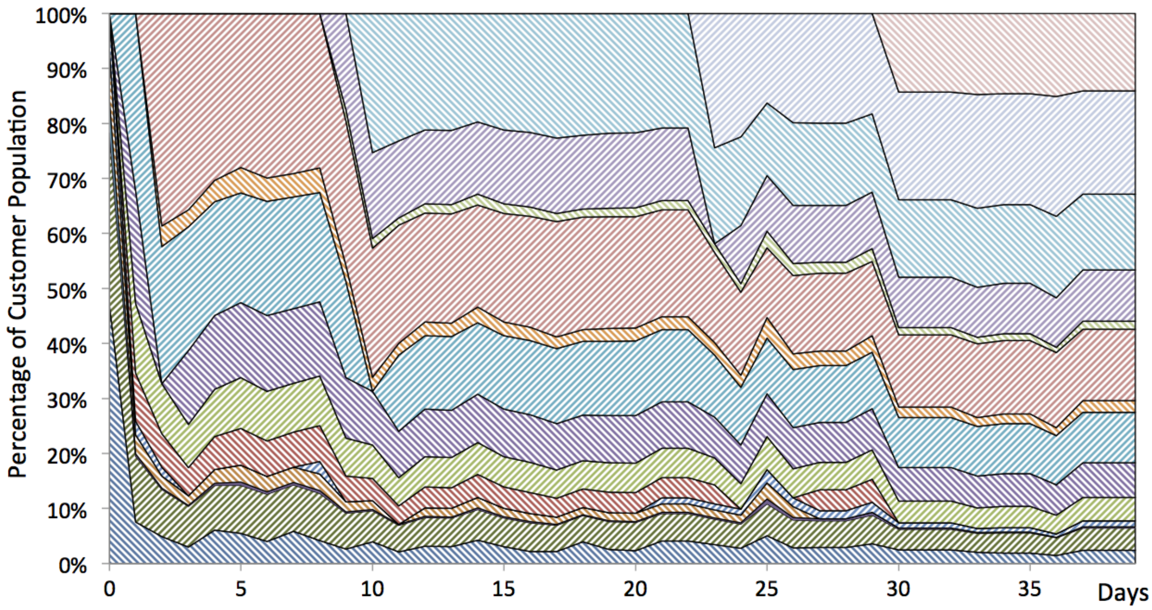


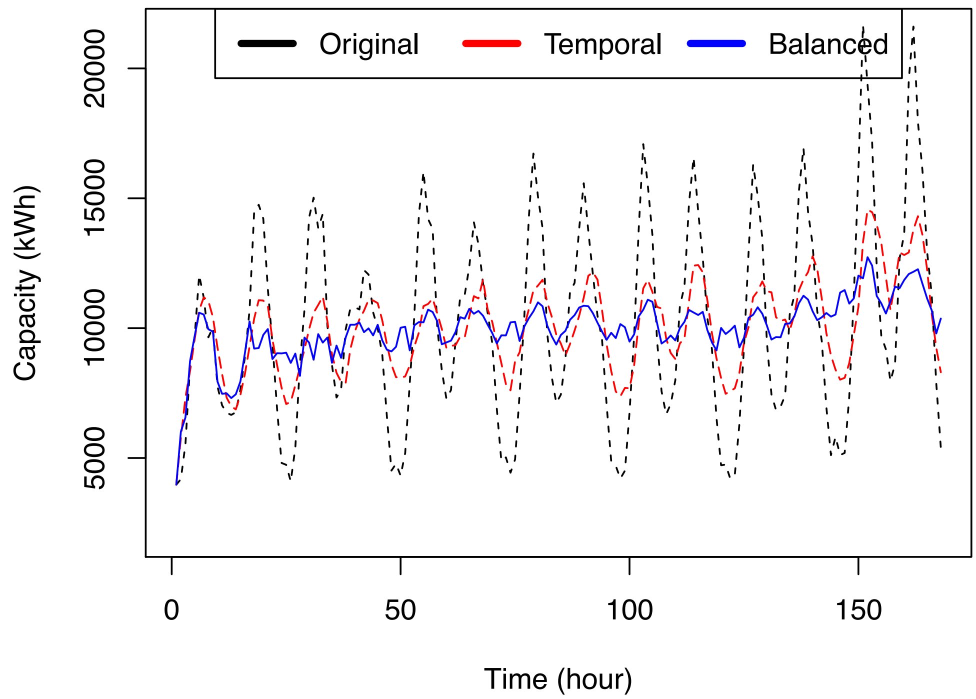
Figure 4.6: Allocation by percentage of a population of 30000 residential consumers as new tariffs enter the market over a period of 40 days.

shifts enabled (long-dashed red line), and the shifted capacities with only balancing shifts enabled (solid blue line). Figure 4.7b highlights the reduced variance in the shifted consumption patterns using box plots and also shows the effect of combining temporal and balancing shifts. Finally Figure 4.8 illustrates the equivalent adaptation of capacity profiles in a Power TAC tournament in September 2011.

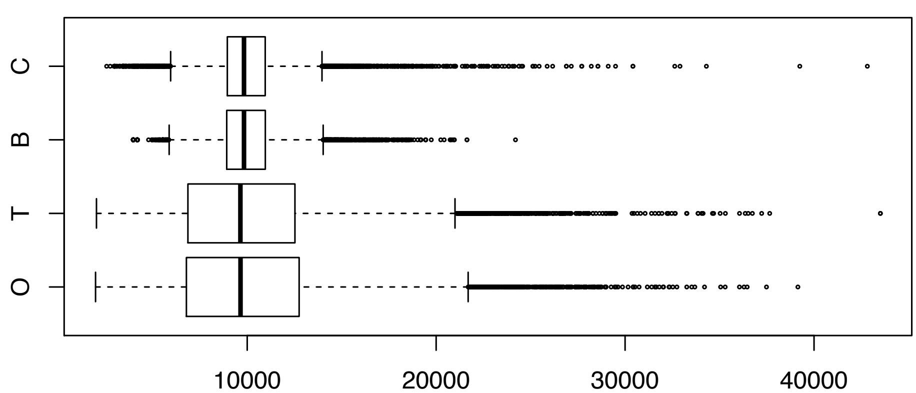
4.3 Chapter Summary

In this chapter, we formalized the decision-making responsibilities of Smart Grid customers as a multiscale decision-making problem along temporal and contextual dimensions. We then introduced our factored customer model representation, a configuration-driven software framework to represent many varieties of Smart Grid customer populations at varying levels of granularity. Customer models instantiated using this framework can be used in Power TAC simulations for tournaments and offline research.

We used the example scenario of a rural electric cooperative to highlight the semi-cooperative relationships between the various agents in the environment. We contributed a stochastic adaptive capacity management algorithm that computes an ϵ-quantal response equilibrium for Smart Grid customer agents in such multi-dwelling scenarios. We demonstrated through simulation



(a) This subfigure shows the raw time series over a representative episode for the original capacities and the temporal and balancing shifting capacities.



(b) This subfigure shows the corresponding box plots for the three series of subfigure (a) and also one labeled C for the temporal and balancing shifts combined.

Figure 4.7: The emergent consumption capacity of the population when they do not shift capacities, use *temporal* shifts, use *balancing* shifts, or both.

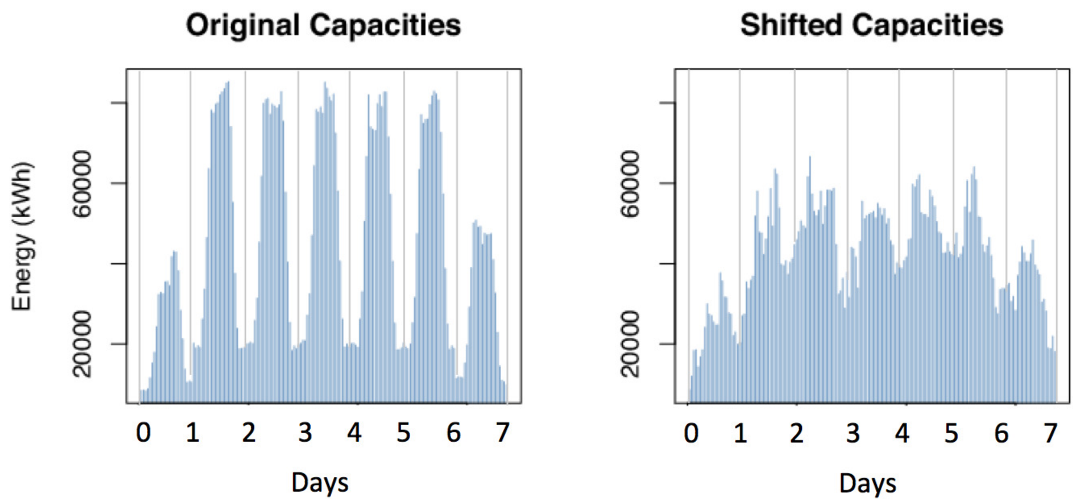


Figure 4.8: Emergent capacity of a factored customer at its original levels (left subfigure) and with adaptive capacity optimization (right subfigure) in a Power TAC tournament.

experiments that this algorithm achieves cost savings for the customers and smoothing of their aggregate capacities without exhibiting herding behavior under typical variable rate tariffs.

In Chapter 2 we developed learning strategies for broker agents and in this chapter we developed adaptive semi-cooperative strategies for customer agents. In the next chapter, we combine principles from these two contributions and also from related work in behavioral game theory to develop learning customer agents.

Chapter 5

Negotiated Learning

This chapter introduces *Negotiated Learning*—a novel approach that we use to develop learning Smart Grid customer agents. Section 5.1 formulates the *variable rate tariff selection* problem, which characterizes the class of problems that can be addressed by Negotiated Learning. In Section 5.2, we identify the dynamic multiagent structure of the problem, define a representation that captures the relevant structure, and present an algorithm that exploits such structure to solve the problem. Section 5.3 formulates a second Smart Grid customer agent problem, *capacity aggregate management*, that also exhibits similar structure and can be addressed by Negotiated Learning. In Section 5.4, we explore the applicability of Negotiated Learning beyond the Smart Grid domain. We preview some experimental results in this chapter, but we mostly defer the experimental analysis of Negotiated Learning to Chapter 6.

5.1 *Problem*: Variable Rate Tariff Selection

The introduction of supplier competition through deregulation of retail markets encourages novel tariff structures and provides customers with more tariff choices so that they can select tariffs best suited for their specific consumption behavior and risk appetite [Block et al., 2010]. However, the resulting *tariff selection* problem, *i.e.*, periodically selecting amongst the set of competitive tariffs, is difficult when prices are allowed to change rapidly.

The problem is further complicated when we assume that the price changes for a particular tariff are published in real-time only to those customers that are currently subscribed to that tariff, thus making the prices partially observable when selecting amongst the tariffs. This assumption

is grounded in the real world and observable already in emerging retail markets such as those in Pennsylvania, New York and Texas.[1]

In formulating the *variable rate tariff selection* problem, we assume deregulated markets where brokers compete to acquire portfolios of customers. Each broker offers one or more tariffs to which customers can *subscribe*, *i.e.*, accept without modification of the tariff contract. The contract includes various terms and fees including one or more rate specifications.

Definition 5.1: The **metering period** is the length of time between successive observations of a customer's cumulative consumption.

Definition 5.2: The **advance notice window** is the length of time between when a dynamic tariff price is communicated to a customer and when that price becomes applicable.

Definition 5.3: A **variable rate** specifies that the dynamic price to be charged for a metering period is communicated to subscribed customers at the start of some advance notice window.

For example, with a 1-hour metering period and 4-hour advance notice window, the price to be charged for 5-6pm is communicated to the customer at 1pm, for 6-7pm at 2pm, and so on. Metering periods and advance notice windows vary widely amongst real-world tariffs; they are usually measured in days or hours for residential customers and hours or minutes for commercial customers. In the following discussion and experiments, without loss of generality, we assume a 1-hour metering period and no advance notice.

A customer must always be subscribed to one tariff in order to maintain electricity supply. The customer can select a different tariff, *i.e.*, switch their tariff subscription, at any time. The switch is effective starting at the next metering period.

Definition 5.4: The **switching cost**, c_s, is a one time cost charged to a customer each time the customer switches from one selection to another. This cost represents actual fees charged by the distribution utility and also the effort involved on the customer's part to execute the switch.

The prices conveyed through a variable rate specification are a key component of the uncertainty in evaluating which tariff is best suited for a particular customer. Since prices evolve over time, the customer benefits from reevaluating the tariffs continuously and thus tariff selection is better described as a sequential decision process rather than a singular event.

We define the resulting *tariff selection decision process* over the discrete time sequence $\mathcal{T}$. Given a set of tariffs, $\mathcal{Z}$, a policy π of the process is a map of tariff *subscriptions* over time:

$$\pi : \mathcal{T} \to \mathcal{Z} \tag{5.1}$$

[1] Appendix E provides references to samples of tariff contracts which indicate that the rate is variable and offer no easily accessible means for potential customers to observe the rates.

We assume that the decision process is given a set of *capacity forecasts*, $\mathcal{Y}$, by the environment. Each forecast $\hat{y} \in \mathcal{Y}$ is a map $\mathcal{T}_t^{t+H} \to \mathbb{R}^+$ and represents an expected consumption pattern over a tariff evaluation horizon, H.

At time t, tariff $z \in \mathcal{Z}$ specifies a price $p_z(t)$. Then, let $p^\pi(t)$ be the price specified by the tariff $z^\pi(t)$, *i.e.*, the tariff to which the customer is subscribed at time t. The goal of the agent is to minimize the cumulative costs over the sequence of observed capacity levels $y(t)$, given the capacity forecasts $\mathcal{Y}$:

$$\min_{\pi} \sum_{t \in \mathcal{T}} p^\pi(t)\, y(t) + \mathbb{1} c_s(t) \tag{5.2}$$

This definition of the problem is similar to the nonstochastic or adversarial multi-armed bandit problem, where a player must choose one of several slot machines–*bandits*–to play at each time step without making any statistical distribution assumptions for the rewards from each bandit [Auer et al., 1995]. For this problem, the EXP3 family of algorithms provide strategies for balancing exploration and exploitation using exponential-weighting to achieve optimal performance bounds. However, as we show in experimental results, our Negotiated Learning algorithm produces significantly better results than EXP3/EXP3.P/EXP3.S when applied to the variable rate tariff selection problem because our approach assumes and exploits the specific multiagent structure of the problem that we explain in Section 5.2.1.

5.2 Negotiated Learning

We now describe how Negotiated Learning allows a self-interested sequential decision-making agent to periodically select amongst a variable set of *entities* (*e.g.*, tariffs) by negotiating with other agents in the environment to gather information about dynamic partially observable entity *features* (*e.g.*, tariff prices) that affect the entity selection decision.

Section 5.2.1 describes the multiagent structure of the variable rate tariff selection problem that is exploited by Negotiated Learning. Section 5.2.2 formulates the problem as a *Negotiable Entity Selection Process* (NESP). Section 5.2.3 describes how a Negotiated Learning agent uses our ATTRACTION-BOUNDED-LEARNING algorithm to determine when to acquire which information from which other agents to help make its entity selection decisions.

To aid in positioning the components of Negotiated Learning relative to each other, we draw an analogy to Reinforcement Learning. The NESP is the equivalent of an MDP in Reinforcement Learning—it serves as the representation for formulating Negotiated Learning problems. ATTRACTION-BOUNDED-LEARNING is the equivalent of Q-LEARNING or SARSA(λ).

We envision Negotiated Learning as being applicable to a class of problems beyond the Smart Grid domain. In Section 5.2.2, we generalize the terminology for problem formulation and, in Section 5.4, we identify some other problems in the class. However, this thesis focuses on validating Negotiated Learning in Smart Grid agents—the variable rate tariff selection problem provides a concrete case study of the target class of problems.

5.2.1 Negotiable Partial Observability

The uncertainty in the expected costs for the tariff choices, $\mathcal{Z}$, in the variable rate tariff selection problem can be attributed to three causal factors:

1. **Price Imputation Uncertainty**: When prices in variable rate tariffs are published only to customers that subscribe to the tariff, it is possible that for some tariffs the only historical price information available to the customer agent is some initial or reference price. Then, the agent must apply an *imputation method* (IM) to estimate any missing prices.

2. **Price Forecasting Uncertainty**: Even if perfect information about past prices is available, the agent must still apply a *forecasting method* (FM) to estimate how the prices will evolve over the tariff evaluation horizon, H. Trivially, the agent could assume that the prices are equal to the last observed value, but in many domains including tariff prices, this would be a poor assumption given that prices correlate with various factors such as daily cycles.

3. **Capacity Forecasting Uncertainty**: Forecasts of customer capacity typically increase in uncertainty as the time span of the forecast increases. Moreover, while the difference in cost between tariffs per unit capacity (*e.g.*, kWh) depends only on the tariff price, the total cost for a time step obviously depends on the total metered units of capacity. Therefore, if the capacity aggregate for a certain period is very low, switching to a better tariff is not as compelling during that period.

Since tariff selection is a forward-looking optimization, only the uncertainty in price forecasting and capacity forecasting affect the decision. For the variable rate tariff selection case study, we assume that the capacity forecasting uncertainty is difficult to mitigate as it stems from factors that the agent cannot observe or control. However, price forecasts are often highly dependent on price histories, which raises the question of whether the agent can improve its price forecasts, and therefore its tariff selection decisions, by mitigating the price imputation uncertainty.

We observe that since tariffs are published contracts, the prices for a particular variable rate specification are the same for all potential customers. So, even though the prices for tariffs that

the customer is not subscribed to are hidden from the customer agent, the agent has the ability to potentially acquire current *price samples* or entire *price histories* from other customers who are subscribed to those tariffs. Thus, it is possible for the population of customers to cooperatively pool their information and decrease the amount of hidden information for each of them.

However, we assume a more realistic model where each customer is *self-interested* and *semi-cooperative*; *i.e.*, each customer needs to be incentivized to share their information. Incentives can take many forms such as in-kind exchange of information, credits for future use, or cash payments. If our decision-making agent wants to acquire information from another customer, it must *negotiate* with that customer for that information. We can intuitively expect, as we also demonstrate in our experiments, that *learning from this negotiated information* can significantly reduce the price imputation uncertainty.

A customer agent can view the population of other customers as a *multiagent oracle*, albeit an incomplete one since some information is hidden from all customers. We refer to this semi-cooperative multiagent structure as **negotiable partial observability**. In Section 5.2.2, we enrich the definition of the variable rate tariff selection problem to explicitly represent this structure. In Section 5.2.3, we describe in detail how our ATTRACTION-BOUNDED-LEARNING algorithm addresses the price imputation, price forecasting, and capacity forecasting uncertainties.

5.2.2 Negotiable Entity Selection Process

In this section, we first define a *negotiable entity selection* problem, using a series of definitions. We then define the general representation of a Negotiable Entity Selection Process (NESP). Lastly, we instantiate a specific NESP representation for the variable rate tariff selection case study.

5.2.2.1 Negotiable Entity Selection Problem

Definition 5.5: An **entity** is defined by one or more **entity features** that contribute to the utility value of that entity as perceived by a decision-making agent in an *entity selection* problem. The value of an entity feature may be static or dynamic.

Definition 5.6: An **entity selection** problem for a decision-making agent requires the agent to select exactly one entity at each time step.

Definition 5.7: A **partially observable entity** has at least one entity feature that is not fully observable to the decision-making agent in an entity selection problem.

Definition 5.8: A **negotiation** is a communication executed over multiple time steps by two semi-cooperative agents where (i) the first agent requests the observed value of an entity feature

from the second agent in exchange for a payment equal to the *negotiation cost*, and (ii) the second agent optionally responds with the requested observation or a declination.

Definition 5.9: The **negotiation cost** for an entity feature is determined by the agent responding to the negotiation request.

Definition 5.10: A **negotiable entity** is a partially observable entity in a distributed agent environment where (i) the hidden entity features are perceived identically by all agents to which they are observable, and (ii) the perceived entity features can be communicated from one agent to another through a negotiation.

Definition 5.11: A **negotiable entity selection** problem for a decision-making agent requires the agent to select exactly one negotiable entity at each time step.

5.2.2.2 General NESP Representation

Definition 5.12: A **Negotiable Entity Selection Process** is a structured representation of a negotiable entity selection problem for a decision-making agent. It is defined as:

$$\langle \mathbf{K}, \mathcal{Z}, \varphi(\mathbf{S}, \mathcal{F}), \mathbf{N}, \mathcal{A}, \mathrm{T}, \mathrm{R} \rangle$$

where:

- $\mathbf{K}$ is the **agent class model** defined by an agent classification map $\mathcal{I} \to \mathcal{K}$ where $\mathcal{I} = \{i_j\}_{j=1}^{|\mathcal{I}|}$ is the set of agents in the environment and $\mathcal{K} = \{k_j\}_{j=1}^{|\mathcal{K}|}$ is the set of **agent classes**.

- $\mathcal{Z}$ is the time-varying set, defined as $\mathcal{Z}(t) = \{z_j\}_{j=1}^{|\mathcal{Z}(t)|}$, of negotiable entities from which the decision-making agent must select exactly one at each time step t. The representation of each z is a domain-dependent function of the entity features.

- $\varphi(\mathbf{S}, \mathcal{F})$ is the **state transform function** on the **state model** $\mathbf{S}$ and the set of **state features** $\mathcal{F} = \{f_j\}_{j=1}^{|\mathcal{F}|}$. At each t, $\mathbf{S}$ contains the current state $s(t)$. φ is defined such that $\varphi(s(t), \mathcal{F})$ generates a set of states of size $|\mathcal{F}|$. Each element $\varphi(s(t), f_j)$ of the generated set is the **transformed state** that is obtained if the entity feature f_j is observed through negotiation.

- $\mathbf{N}$ is the **negotiation model**, which is defined as a bipartite graph, $\varphi(\mathbf{S}, \mathcal{F}) \to \mathcal{K}$. A sample negotiation model is illustrated in Figure 5.1. Each edge of the graph represents a **negotiation action**—it connects a transformed state $\varphi(s(t), f_j)$ to an agent class $k_{j'}$. Each edge carries a triple of **negotiation parameters** (c, τ, x), where c is the **cost of information**, τ is the **interval to information**, and x is the **probability of information**, with the intuitive

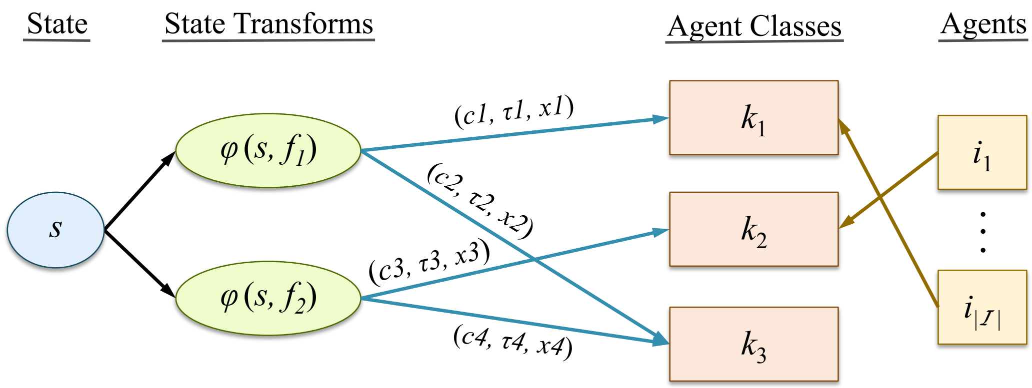


Figure 5.1: Example negotiation model $\mathbf{N}$ with $|\mathcal{F}|$=2 and $|\mathcal{K}|$=3 where each edge of the bigraph is a *negotiation action* $\in \mathcal{A}_1(t)$ with (c, τ, x) parameters.

understanding that if an agent in $k_{j'}$ responds to a negotiation request quickly (low τ) with high reliability (high x), then the cost c of the negotiation is likely to be higher.

- $\mathcal{A}$ is the time-varying set of actions defined as $\mathcal{A} = \mathcal{A}_1(t) \cup \mathcal{A}_2(t)$. $\mathcal{A}_1(t)$ is the set of **negotiation actions**, *i.e.*, edges in the negotiation model given the current state $s(t)$. $\mathcal{A}_2(t)$ is the set, of size $|\mathcal{Z}(t)|$, of **entity selection actions**. At each t, the decision-making agent executes zero or more actions from $\mathcal{A}_1(t)$ and selects exactly one action from $\mathcal{A}_2(t)$.

- T is the state transition function, which is dependent on the entity selection actions and any executed negotiation actions at each t.

- R is the reward function, which is dependent on the entity selection actions at each t and the costs resulting from any completed negotiation actions.

Generally, in a dynamic stochastic multiagent environment such as the Smart Grid, the exact transition function and reward function are unknown, or hidden from the decision-making agent. However, it is unclear whether some or all of the agent class model, $\mathbf{K}$, and the negotiation model, $\mathbf{N}$, will generally be known or hidden. In our experimental analysis, we consider scenarios where the agent classification map in $\mathbf{K}$ and the negotiation parameters in $\mathbf{N}$ are hidden.

The Negotiable Entity Selection Process is a representation that enables the decision-making agent to exploit the structure of negotiable partial observability described in Section 5.2.1. In particular, by separating the multiple time step negotiation actions from the per time step entity selection actions, and allowing for their simultaneous execution, the representation allows the

agent to separately control its exploration and exploitation behaviors. We demonstrate this aspect in our ATTRACTION-BOUNDED-LEARNING algorithm in later sections, but first we focus on presenting a deeper understanding of the NESP representation by instantiating it for the variable rate tariff selection problem and another problem that we introduce in Section 5.3.

5.2.2.3 Instantiated NESP Representation

To develop an NESP instantiation for the variable rate tariff selection case study, we consider a scenario that we use in our experiments in Chapter 6. We define the instantiated NESP as:

$$\langle \mathbf{K}, \mathcal{Z}, \varphi(\mathbf{S}, \mathcal{F}), \mathbf{N}, \mathcal{A}, \mathrm{T}, \mathrm{R} \rangle$$

where:

- The set of agents $\mathcal{I}$ contains 60 agents of various configurations that are mapped into two agent classes: $\mathcal{K} = \{$*Desirable, Undesirable*$\}$. The agent classification map $\mathcal{I} \rightarrow \mathcal{K}$ in $\mathbf{K}$ is hidden from the decision-making agent.

- The fixed set of negotiable entities $\mathcal{Z} = \{$T1, T2, T3, T4, T5$\}$ contains 1 fixed rate tariff and 4 variable rate tariffs. Each tariff has two negotiable entity features: $\{$*PriceSample, PriceHistory*$\}$. A negotiation request for a *PriceSample* $p_z(t)$ of a tariff z can be fulfilled by the responding agent only if that agent is currently subscribed to z, whereas a request for *PriceHistory*, if fulfilled, yields all the price values known to the responding agent.

- The state model $\mathbf{S}$ maintains the known prices for each tariff in $\mathcal{Z}$. The prices may be obtained through subscribing to the tariff or through negotiation. The set of state features $\mathcal{F}$ is the cross product of the 5 tariffs in $\mathcal{Z}$ crossed with their 2 negotiable entity features each. Thus, $\varphi(s(t), \mathcal{F})$ generates $5 \times 2 = 10$ transformed states.

- The negotiation model $\mathbf{N}$ is hidden. If it were assumed to be known, it would reveal that agents in the *Undesirable* class exhibit relatively worse (c, τ, x) values in their negotiations compared to agents in the *Desirable* class.

- There are 9 negotiation actions in $\mathcal{A}_1(t) = \mathcal{F} \setminus (z(t), \textit{PriceSample})$, *i.e.*, one for each state feature in $\mathcal{F}$ except the one for *PriceSample* of the currently subscribed tariff $z(t)$ since its price is already known. The set of 5 entity selection actions in $\mathcal{A}_2(t)$ corresponds to $\mathcal{Z}$.

- The transition function T is hidden from the agent.

- The reward function R is defined by the per time step reward:

$$r(t) = p_{z(t)}(t)\, y(t) + \mathbb{1}c_s(t) + c_N(t) \tag{5.3}$$

where $p_{z(t)}(t)$ is the price at t of the selected tariff $z(t)$, $y(t)$ is the agent's realized capacity, $\mathbb{1}c_s(t)$ is the tariff switching cost if applicable, and $c_N(t)$ are the negotiation costs.

5.2.3 ATTRACTION-BOUNDED-LEARNING

Definition 5.13: The **Attraction** of an entity is defined by the triple (μ, β^+, β^-), whose elements are interpreted as the mean, upper bound, and lower bound on some domain-dependent measure of that entity's attractiveness. An Attraction is denoted V and $V.\mu$ and $V.\beta^{\pm}$ are in $\mathbb{R}$.

Definition 5.14: The **canonical Negotiated Learning problem** is a negotiable entity selection problem where the negotiable entities have dynamic entity features.

ATTRACTION-BOUNDED-LEARNING exploits the structure exposed by a Negotiable Entity Selection Process to address the challenges posed by dynamic entity features in a negotiable entity selection problem. Specifically, the algorithm focuses on managing simultaneous exploration and exploitation of the negotiable entities. This is in contrast to other algorithms such as EXP3 and Q-LEARNING which generally assume that exploration-exploitation is an inherent tradeoff and therefore devise a strategy to balance that tradeoff.

While the algorithm is applicable to any problem that is defined as a Negotiable Entity Selection Process, we continue to use the tariffs/prices terminology from the variable rate tariff problem instead of the generic entities/features terminology. While this section summarizes the salient aspects of the algorithm, we defer some discussion of the tuning parameters to Section 6.2 where we analyze them with the additional context of sensitivity experiments.

The overall algorithm forms a three-layered learning process:

1. *Learning from negotiated information*: Price samples and histories obtained through negotiation are used in a set of price imputation methods and the resulting imputed price series are used in price forecasting methods to aid in tariff selection as we describe below.

2. *Learning the negotiation model*: If the negotiation model $\mathbf{N}$ in the NESP representation is hidden, the agent's history of negotiations is used to estimate the (c, τ, x) negotiation parameters for the edges in the bipartite graph of $\mathbf{N}$.

3. *Learning the agent classification map*: Assuming the set of agents $\mathcal{I}$ and the set of agent classes $\mathcal{K}$ are known, but the map $\mathcal{I} \rightarrow \mathcal{K}$ in $\mathbf{K}$ is hidden, agents in $\mathcal{I}$ are dynamically mapped to $\mathcal{K}$ based on past negotiations.

The first layer is summarized in Algorithms 5.1-5.3. The second and third layers are included in Algorithm 5.3 and described in some additional detail within the text of this section.

Algorithm 5.1, ATTRACTION-BOUNDED-LEARNING, is activated at each t with the state update $s(z_t^\pi)$ determined by its current entity selection z_t^π. Internally, it maintains the following state in addition to the NESP:

- $\mathcal{V}$, a set of current Attractions for the tariff entities in $\mathcal{Z}$
- $\mathcal{N}$, a set of current (*i.e.*, initiated but incomplete) negotiations
- z_t^π, the previously selected tariff entity

Additionally, the algorithm is given the following tuning parameters:

- **Attraction update weights** $\{\omega_e, \omega_b\}$ in $\mathbb{R}[0, 1]$
- **Attraction bounds decay factor** $\lambda \in \mathbb{R}^+$
- **Negotiation budget factor** $\gamma \in \mathbb{R}[0, 1]$
- **Attraction benefit threshold** $\xi \in \mathbb{R}^+$

5.2.3.1 Management of Exploitation

- Each invocation of the algorithm first subsumes the state update $s(z_t^\pi)$ into the current state model, initializes the tariff selection decision z_{t+1}^π to the current tariff selection, initializes a **negotiation budget** ψ to 0, and then obtains an updated set of non-negotiable entity features, the current capacity forecasts $\mathcal{Y}$ in this case study, from the environment.

- The Attraction V^π for the current tariff is updated according to Algorithm 5.2: ABL-COMPUTE-ATTRACTION, described below. Then for all other tariffs z in the tariff entity choices $\mathcal{Z}$, we first check to see if there is a current negotiation for features of z in $\mathcal{N}$. If

Algorithm 5.1 **ATTRACTION-BOUNDED-LEARNING**$(t, s(z_t^\pi))$

State: $(\mathcal{V}, \mathcal{N}, z_t^\pi)$
Parameters: $(\omega_e, \omega_b, \lambda, \gamma, \xi)$
Return: z_{t+1}^π
$\mathbf{S} \leftarrow \mathbf{S} \cup s(z_t^\pi)$
$z_{t+1}^\pi \leftarrow z_t^\pi$
$\psi \leftarrow 0$
$\mathcal{Y} \leftarrow$ **Env.GetFeatureForecasts**(t)
$V^\pi \leftarrow$ **ABL-COMPUTE-ATTRACTION**$(z_t^\pi, \mathcal{V}[z_t^\pi], \mathcal{Y}, \omega_e, \lambda)$
for z in $\mathcal{Z} \setminus z_t^\pi$ **do**
    $\omega \leftarrow \omega_b$
    $n \leftarrow \mathcal{N}[z]$
    if $n \neq$ null **and** n.*success* = **true then**
        UpdateEntityValues$(\mathbf{S}, z, n)$
        $\omega \leftarrow \omega_e$
    end if
    $V_z \leftarrow$ **ABL-COMPUTE-ATTRACTION**$(z, \mathcal{V}[z], \mathcal{Y}, \omega, \lambda)$
    if $V_z.\beta^+ > V^\pi.\beta^+$ **or** $V_z.\beta^- > V^\pi.\beta^-$ **then**
        $\mathcal{Z}_u \leftarrow \mathcal{Z}_u \cup z$
        if $(V_z.\beta^+ - V^\pi.\beta^+) > (\psi/\gamma)$ **then**
            $\psi \leftarrow \gamma\,(V_z.\beta^+ - V^\pi.\beta^+)$
        end if
    end if
    if $V_z.\mu > V^\pi.\mu + \xi$ **then**
        $z_{t+1}^\pi \leftarrow z$
    end if
end for
$\mathcal{N} \leftarrow$ **ABL-INVOKE-NEGOTIATIONS**$(\mathcal{N}, \mathcal{Z}_u, \psi)$

yes, and the corresponding negotiation has completed successfully, then the price samples or histories obtained through negotiation are incorporated into the state model $\mathbf{S}$.

- That negotiated information is also used to recompute the Attraction V_z for tariff entity z using the **experience update weight** ω_e, instead of the **belief update weight** ω_b. If the upper or lower confidence bound, β^+ and β^-, for z's Attraction is higher than that of the current tariff, it is added to the set of *uncertain tariffs*, $\mathcal{Z}_u$, to be considered for negotiation.

- If the Attraction mean $V_z.\mu$ is greater than that of the current tariff plus the Attraction benefit threshold ξ, then the corresponding tariff z is marked for selection at time $t + 1$.

Algorithm 5.2 **ABL-COMPUTE-ATTRACTION**$(z, V_z, \mathcal{Y}, \omega, \lambda)$

Parameter: κ
Return: V_z
Outcomes $\leftarrow \emptyset$
for i in Γ_i **do**
  $\overrightarrow{h} \leftarrow$ **ImputeFeatureValues**(z, i)
  for j in Γ_p **do**
    $\overrightarrow{p} \leftarrow$ **PredictEntityValues**$(j, \overrightarrow{h})$
    for $\overrightarrow{y}$ in $\mathcal{Y}$ **do**
      for l in $\mathcal{L}$ **do**
        Outcomes $\leftarrow$ *Outcomes* $\cup\ (\overrightarrow{p}[1..l] \cdot \overrightarrow{y}[1..l])/l$
      end for
    end for
  end for
end for
$V_z.\mu \leftarrow (1-\omega)\ V_z.\mu + \omega$ **Mean**(*Outcomes*)
$\beta^* \leftarrow (1+\lambda)\ \kappa$ **StdDev**(*Outcomes*)
$V_z.\beta^+ \leftarrow (1-\omega)\ V_z.\beta^+ + \omega\ (V_z.\mu + \beta^*)$
$V_z.\beta^- \leftarrow (1-\omega)\ V_z.\beta^- + \omega\ (V_z.\mu - \beta^*)$

- Lastly, the uncertain tariffs are evaluated for negotiation by invoking Algorithm 5.3: ABL-INVOKE-NEGOTIATIONS, described below. The negotiations are constrained by a budget ψ that is computed as the product of the negotiation budget factor γ and the maximum expected benefit $V_z.\beta^+ - V^\pi.\beta^+$ over all $z \in \mathcal{Z} \setminus z_t^\pi$.

5.2.3.2 Computation of Attractions

Attractions are key to our approach because they capture the uncertainties in price imputation, price forecasting, and capacity forecasting. Algorithm 5.2 describes their computation:

- We assume availability of a library of domain-dependent **imputation methods**, Γ_i, that fill in missing historical prices, and a library of **forecasting methods**, Γ_p. We generate one *imputation* using each imputation method in Γ_i, and then generate a set of price forecasts for each imputation using each forecasting method in Γ_p.

  For example, in our variable rate tariff selection case study, we use the following general imputation methods and domain-specific forecasting methods.

Imputation Methods -

Hidden values in the price history of each tariff entity are estimated using known prices that may have been observed directly by the agent as a subscriber to the tariff or obtained by the agent through prior negotiations:

1. *Global Mean*: Set all hidden prices equal to the mean of all known prices.
2. *Carry Forward*: Set each hidden price equal to the prior known price.
3. *Back Propagation*: Set each hidden price equal to the next known price, or to the prior known price if all later prices are hidden.
4. *Interpolation*: Each contiguous sequence of hidden prices is assigned using linear interpolation from the prior known price to the next known price.

Forecasting Methods -

These methods assume that tariff prices have a strong correlation at *lag 24*, *i.e.*, their prices for the same hour the previous day, and that they are autoregressive.

1. *Lag 24*:

$$p_t = p_{t-24} + \varepsilon,\ \varepsilon \sim N(0, \sigma^2) \tag{5.4}$$

2. *AR(1)*: An autoregressive series of order 1 (see Appendix C):

$$p_t = \mu + \phi\, p_{t-1} + e_t \tag{5.5}$$

3. *ARMA(1,1)*: An autoregressive moving average series of order $(1, 1)$:

$$p_t = \mu + \phi\, p_{t-1} + e_t + \theta\, e_{t-1} \tag{5.6}$$

4. *SARMA(1,1)×(1,1)*$_{24}$: An autoregressive moving average series of order $(1, 1)$ with a seasonal component also of order $(1, 1)$:

$$p_t = \mu + \phi\, p_{t-1} + \Phi\, p_{t-24} + e_t + \theta\, e_{t-1} + \Theta\, e_{t-24} + \Theta\,\theta\, e_{t-25} \tag{5.7}$$

- Capacity forecasts have higher uncertainty farther into the future, so we give more weight to forecast values for the near future. We do this by choosing a set of **lookahead windows**, $\mathcal{L}$, all less than the tariff evaluation horizon, H. For each capacity forecast $\vec{y}$ in $\mathcal{Y}$, for each lookahead window in $\mathcal{L}$, for each price forecast in $\Gamma_i \times \Gamma_p$, we compute one **outcome**. In the variable rate tariff selection case study, each outcome represents the average hourly cost

Algorithm 5.3 **ABL-INVOKE-NEGOTIATIONS**$(\mathcal{N}, \mathcal{Z}_u, \psi)$

State: (B)
Parameter: ω_α
for n in $\mathcal{N}$ **do**
  if n.*status* = Completed **then**
    $\mathrm{B}.\mathcal{Z}[n.i] \leftarrow n.z$
    $\mathbf{N}[\mathbf{K}[n.i]].(c,t,p) \leftarrow (1-\omega_\alpha)\,\mathbf{N}[\mathbf{K}[n.i]].(c,t,p) + \omega_\alpha\, n.(c,t,p)$
  end if
  $\mathrm{B}.\mathcal{K}[n.i] \leftarrow$ **ReclassifyNeighbor**$(n.i)$
  $\mathcal{N} \leftarrow \mathcal{N} \setminus n$
end for
$\mathcal{A}_1 \leftarrow \varphi(\mathcal{Z}_u, \mathcal{F}) \times \mathcal{K}$
$\mathcal{N}^* \leftarrow$ **ZeroOneProgram**$(\mathcal{A}_1, \psi)$
$\mathcal{N} \leftarrow \mathcal{N} \cup \mathcal{N}^*$
for n in $\mathcal{N}^*$ **do**
  $i \leftarrow$ **SelectNeighbor**$(n.k)$
  Env.InitiateNegotiation(i, n)
end for

charged to the agent under the given capacity forecast $\vec{y}$ if the agent were subscribed to the tariff z associated with the Attraction V_z that is being computed.

- We thus collect $|\Gamma_i| \times |\Gamma_p| \times |\mathcal{Y}| \times |\mathcal{L}|$ real-valued outcomes. The statistical mean of the distribution of outcomes is used to update the Attraction mean $V_z.\mu$. The standard deviation of the distribution is used to update the Attraction bounds $V_z.\beta^{\pm}$. The Attraction bounds decay factor λ, if > 0, enables the bounds to diverge over time to trigger exploration. The parameter κ determines the confidence interval to be used in computing the bounds.

Intuitively, the Attraction bounds, (β^+, β^-), are used to capture the uncertainty in the agent's belief about a certain entity, whereas the mean, μ, is used to capture the expectation. The bounds determine *when and which negotiations to undertake*, based on the potential benefit of selecting an alternate entity, thus controlling the *exploration* process. The mean determines *which entity to select*, thus controlling the *exploitation* process. By separating the two concerns of exploration-exploitation into different signals, we are able to control exploration costs more intelligently, which is of critical importance when selecting the wrong entity can be very expensive.

5.2.3.3 Management of Exploration

In addition to the negotiation model, $\mathbf{N}$, which includes the (c, τ, x) *parameters* for each negotiation action, the agent maintains a **neighbor model** B. The neighbor model contains the agent's

beliefs about which tariffs its neighbors are subscribed to and about the agent class model, $\mathbf{K}$; *i.e.*, (i) $\mathbf{B}.\mathcal{Z} = \mathcal{I} \to \mathcal{Z}$, and (ii) $\mathbf{B}.\mathcal{K} = \mathcal{I} \to \mathcal{K}$.

- Algorithm 5.3 first uses information from any completed negotiations in $\mathcal{N}$ to update the neighbor-tariff map $\mathbf{B}.\mathcal{Z}$ for the neighbor identified in the negotiation.

- It then applies a weighted update to the negotiation parameters for the neighbor's agent class. The cost c of the negotiation is determined by the neighbor and x is 0 or 1 to indicate negotiation failure or success.

- If the agent classification map is unknown, then the neighbor is reclassified in $\mathbf{B}.\mathcal{K}$ using the negotiation's realized (c, τ, x) value and domain-specific heuristics.

- $\mathcal{A}_1(t)$ is the current set of *negotiation actions* derived as the cross product of the state transforms on the uncertain tariffs $\varphi(\mathcal{Z}_u, \mathcal{F})$, crossed with the agent classes $\mathcal{K}$. We then obtain a set of desired negotiations, $\mathcal{N}^*$, by solving a zero-one program over $\mathcal{A}_1(t)$ with the goal of maximizing the expected information value and the constraints that:

  - the total cost of the negotiations is no more than the negotiation budget ψ, and
  - no more than one action is chosen for a particular state transform.

  Algebraically, let $\vec{M}$ be an array of size $|\mathcal{Z}_u| \times |\mathcal{F}| \times |\mathcal{K}|$. Each element m_{ufk} in $\vec{M}$ represents the **information value** of a given negotiation action, $a_{\{u,f,k\}}$, where $u \in \mathcal{Z}_u$, $f \in \mathcal{F}$, and $k \in \mathcal{K}$. Let c_{ufk}, τ_{ufk}, and x_{ufk} be the *cost*, *interval* and *probability* from the (c, τ, x) parameters of the negotiation edge representing $a_{\{u,f,k\}}$ in the negotiation model's bipartite graph. Then, we define each element in $\vec{M}$ as the probability-weighted change in the Attraction bounds, discounted for the time needed for the negotiation:

$$m_{ufk} = \underbrace{(1-\omega_e)^{t+\tau_{ufk}}}_{\text{time discount}} \; \underbrace{x_{ufk}}_{\text{probability}} \; \underbrace{|\beta^+_{uf}(t) - \beta^+_{uf}(t+\tau_{ufk})|}_{\text{change in bounds}} \tag{5.8}$$

  Let $\vec{W}$ be a binary array of the same dimensions as $\vec{M}$, where $\mathbb{1}_{ufk}$ are the corresponding contained elements such that non-zero values represent negotiations to be initiated. The

zero-one optimization problem is then stated as:

$$\max_{\vec{W}} \sum_{u,f,k} \mathbb{1}_{ufk}\, m_{ufk} \tag{5.9}$$

$$\sum_{u,f,k} \mathbb{1}_{ufk}\, c_{ufk} \leq \psi \tag{5.10}$$

$$\sum_{(u,f)} \mathbb{1}_{ufk} \leq 1, \quad \forall\,(u,f) \tag{5.11}$$

- For each desired negotiation in $\mathcal{N}^*$, with probability $1-\epsilon$ we find the neighboring agent that was most recently mapped to that tariff in $\mathrm{B}.\mathcal{Z}$ and initiate negotiation with that neighbor, and with probability ϵ we choose a neighbor to negotiate with randomly.

In Chapter 6, we present the experimental analysis where we validate the NESP formulation of the variable rate tariff selection case study and the application of the ATTRACTION-BOUNDED-LEARNING algorithm. We also study the impact of the choices for the imputation and forecasting methods and other parameters that govern the performance of the algorithm.

5.3 *Problem*: Capacity Aggregate Management

This section formulates another problem within the Smart Grid domain where Negotiated Learning is applicable. Consider the scenario of a commercial building with many tenants.

Definition 5.15: The **capacity aggregate** of a building is the sum of the capacity from each tenant and the capacity of the building infrastructure and common facilities.

Assume that tenants have autonomy over their capacity patterns and they pay a fixed charge for their usage, or are billed at a fixed rate. The building manager, who is responsible for the capacity aggregate, has a choice amongst various tariffs offered by competitive brokers.

This scenario is approximately equivalent to the rural electric cooperative scenario that we explored in Chapter 4. Similar to the *utility optimizer* there, the building manager here, or an autonomous agent on the manager's behalf, faces the multiscale decision-making problem of choosing a suitable tariff while also managing capacity profiles on a finer timescale. The key difference here is that the building manager has *controllable capacity* at each t, that powers the building infrastructure and common facilities. The capacity value of the controllable capacity at each t is determined by the selected capacity profile $\rho(t)$. Shifting to alternate capacity profiles potentially allows the building manager to better utilize the currently applicable tariff.

To illustrate, assume a *tiered* time-of-use (TOU) tariff where rates are not only higher during peak periods, but how much higher depends on the capacity aggregate values during that period.

Definition 5.16: A **tier threshold** in a variable rate tariff sets the capacity limit, which when crossed results in higher prices. It is dependent on the specific tariff whether crossing the threshold entails going higher or lower than the capacity limit.

For example, assuming an hourly metering period and a tier threshold of 10kWh, if the capacity for a particular hour is less than 10kWh, the rate is $0.12/kWh, whereas if it's greater than 10kWh, the rate is $0.20/kWh. Such tiered rates are common in real world tariffs.

If the building is subscribed to such a tiered TOU tariff, the building manager could benefit from *shifting* the controllable capacity away from the intrinsically preferred *default* profile when the capacity aggregate is expected to cross the threshold into the higher rate tier.

Definition 5.17: A **shifting penalty**, in terms of comfort, convenience, or operating costs, may be incurred by a building manager while on a shifted capacity profile.

Nonetheless, as long as the savings accrued from keeping the capacity aggregate in the lower rate tier are greater than the utility-equivalent of the shifting penalty and one-time *switching costs*, then shifting is a rational decision-theoretic strategy.

Definition 5.18: An **aggregating agent** is responsible for optimizing a capacity aggregate by selecting the capacity profile, $\rho(t) \in \mathcal{P}$, of the controllable capacity under the discretion of that agent. The agent is given a static set of **aggregate components**, $\mathcal{Q}$, where each $Q \in \mathcal{Q}$ contributes $y_Q(t)$ in realized capacity at each time t.

The capacity aggregate $\zeta(t)$ at time t is then:

$$\zeta(t) = \rho(t) + \sum_{Q \in \mathcal{Q}} y_Q(t) \tag{5.12}$$

The goal of the aggregating agent is to minimize the cumulative costs over the sequence of realized capacity aggregate values, given the price of the applicable tariff $p_z(t)$, the shifting penalties c_ρ, and the switching costs c_s:

$$\min_{\pi} \sum_{t \in \mathcal{T}} p_z(t)\zeta(t) + \mathbb{1}c_\rho(t) + \mathbb{1}c_s(t) \tag{5.13}$$

The EXP3 family of algorithms can also be applied here, but as we show in Chapter 6, our Negotiated Learning approach produces significantly better results than EXP3.P by leveraging the multiagent structure of the problem.

NESP Representation

We represent a minimal scenario of this problem as a Negotiable Entity Selection Process:

$$\langle \mathbf{K}, \mathcal{Z}, \varphi(\mathbf{S}, \mathcal{F}), \mathbf{N}, \mathcal{A}, \mathrm{T}, \mathrm{R} \rangle$$

where:

- The set of agents $\mathcal{I}$ contains 3 agents that are mapped 1-to-1 into 3 agent classes: $\mathcal{K} = \{$*Aggregator, StableComponent, VolatileComponent*$\}$. The decision-making agent is the aggregating agent, which maps to the *Aggregator* class. The map entries $\mathcal{I} \to \mathcal{Z}$ in $\mathbf{K}$ for the 2 aggregate component agents into the other 2 classes are unknown.

- The set of negotiable entities $\mathcal{Z}$ contains 2 capacity aggregate entities: $\{$*Default, Shifted*$\}$. How we formulate entities presents the most interesting deviation in representing this problem, compared to variable rate tariff selection. In tariff selection, the entities are the tariffs, which are also the choices for selection. Here, the entities are not the capacity profiles ρ, but instead the capacity aggregates ζ. We define a 1-to-1 correspondence between the profiles and aggregates such that a specific aggregate $\tilde{\zeta}(t)$ is the aggregate obtained by the aggregating agent if it selects the capacity profile $\tilde{\rho}(t) \in \mathcal{P}$ at time t. Each aggregate component has 3 negotiable entity features: $\{$*CapacityHistory, CapacitySample, CapacityForecast*$\}$.

- The state model $\mathbf{S}$ maintains the known capacity values for each aggregate component Q. The agent does not need to maintain the capacity history for the controllable capacity because it does not need to forecast from that history—the agent's capacity profile selection fully controls the future capacity values. The set of state features $\mathcal{F}$ is the cross product of the 2 aggregate components crossed with their 3 negotiable entity features each. Thus, $\varphi(s(t), \mathcal{F})$ generates $2 \times 3 = 6$ transformed states.

- The negotiation model $\mathbf{N}$, if known, reveals that the (c, τ, x) parameters for the *StableComponent* agent are relatively worse than for the *StableComponent* agent. In another deviation from the variable rate election problem, we see here that the aggregating agent does not have a choice of which component agent to negotiate with to obtain the capacity values for particular state feature. Instead, the agent must decide which state features to negotiate for at each time step t.

- There are 6 negotiation actions in $\mathcal{A}_1(t)$ corresponding to the 6 transformed states in $\varphi(s(t), \mathcal{F})$. The set of entity selection actions $\mathcal{A}_2(t)$ corresponds to the two capacity profile choices for $\rho(t)$: $\{$*Default, Shifted*$\}$.[2]

- The transition function T is hidden from the agent.

- The reward function R is defined by the per time step reward:

$$\zeta(t) = \rho(t) + \sum_{Q \in \mathcal{Q}} y_Q(t) \tag{5.14}$$

$$r(t) = p_{z(t)}(t)\, \zeta(t) + \mathbb{1} c_s(t) + c_N(t) \tag{5.15}$$

  where $p_{z(t)}(t)$ is the price at t of the selected tariff $z(t)$, $\zeta(t)$ is the realized capacity aggregate, $\mathbb{1} c_s(t)$ is the tariff switching cost if applicable, and $c_N(t)$ are the costs resulting from any completed negotiation actions.

While the capacity aggregate management problem is presented in the context of a commercial building in this case study, consider that a similar scenario exists in some apartment buildings. Also, within a single home, the capacity aggregate is the sum of capacity from several components such as HVAC (heating, ventilation and air conditioning), clothes washer/dryer, electric water heater, swimming pool pump, *etc*. Some of these components can exhibit autonomy, for example if the HVAC is controlled by a *smart thermostat*. The auto-responsive behavior of the thermostat may also make the HVAC capacities *volatile*, for example in response to weather changes, whereas other components such as lighting and washer/dryer are generally *stable*. The homeowner, as the aggregating agent, may then use the water heater or pool pump as controllable capacity to manage the capacity aggregate.

5.4 Beyond Smart Grid Agents

The introduction of Negotiated Learning in this thesis is focused on application to the challenges of developing learning Smart Grid customer agents. We continue that focus with extensive experimental analysis in Chapter 6. In this section, we briefly explore the applicability of Negotiated Learning to problems beyond the Smart Grid domain.

[2]Note that the aggregating agent only selects $\rho(t)$ and not the aggregates $\zeta(t)$, which are a function of the unknown *realized* component capacities $\{y_Q(t)\}$. The differences between the estimated and realized component capacities, $\hat{y}_Q(t)$ and $y_Q(t)$ respectively, contribute to the uncertainties in each entity $\zeta(t)$, thus making capacity aggregate management a problem of negotiable entity selection.

Section 5.2.3 stated that negotiable entity selection with dynamic entity features is the canonical Negotiated Learning problem. The definition of the negotiable entity selection problem in Section 5.2.2 therefore identifies the class of targeted problems formally; nonetheless, it is easier to gain an intuitive understanding of the class through diverse examples.

We first explain how Negotiated Learning is different from a few well-known problem classes:

- **Expert selection** -

  Expert selection problems are a subset of online learning or regret minimization problems where the decision-making agent is informed at time t of the outcomes or rewards that would have been realized for each of the actions that the agent could have chosen at time $t-1$. In Negotiated Learning, the agent is *not* informed directly by the environment of the *forgone* rewards of the actions not chosen.

- **Ratings systems** -

  Ratings systems generally assume uniform preferences over the agents providing the ratings; *i.e.*, each agent applies the same objective measure of utility for the entity being rated.[3] In Negotiated Learning, we do not assume uniform preferences.

- **Portfolio management** -

  In portfolio management problems, the agent is typically allowed to allocate its resources over any subset of the available choices. Therefore, the agent can select multiple choices simultaneously and therefore obtain information about all the selections continuously. In Negotiated Learning, the agent can only select one entity at each t, so it does not have current information about all choices unless it obtains that information from other agents.

Gaining an understanding of these differences allows us to collect the following criteria for "good" Negotiated Learning problems:

1. **Entity selection**: Exactly one entity must be selected at each t, which is sometimes known as the *exclusive service provider* assumption.

2. **Nonstochastic features**: Entity features must be dynamic. Generally, if the feature is dynamic but stationary, then algorithms that target stochastic bandit problems, such as UCB2 [Auer et al., 2002] and MBIE [Strehl and Littman, 2008] are well suited. However, if the feature is assumed to be non-stochastic or adversarial, as targeted by EXP3, then

[3] It may be helpful to contrast with *recommendation* systems, which typically assume diverse preferences.

the problem is more challenging and better suited for the more sophisticated exploration-exploitation mechanism of ATTRACTION-BOUNDED-LEARNING.

3. **Negotiable features**: The realized values of entity features must be perceived identically by each agent so that observed values can be communicated amongst agents.[4] However, the utility value of the negotiable entity is not assumed to be uniform over all agents. So, even if two agents have identical information about a negotiable entity's features, they may select different entities in their decision-making.

Note that it is certainly possible to represent some problems that don't satisfy all the stated criteria (*e.g.*, static entity features) as Negotiated Learning problems, but then they would not necessarily need or benefit from representation as a Negotiable Entity Selection Process or from the application of ATTRACTION-BOUNDED-LEARNING. Table 5.1 summarizes the key attributes of some problems that satisfy the criteria of good Negotiated Learning problems.

Table 5.1: Problems suitable for Negotiated Learning beyond the Smart Grid domain.

Problem	**Agent**	**Entity**	**Features**	**Neighbor Classes**
TV news	Viewer	News channel	– Speed of coverage – Bias in opinions	– Other viewers – Alternate media
Job search	Candidate	Potential employer	– Current prospects – Employee surveys	– Other candidates – Current employees
Legal advice	Business	Retained counsel	– Domain expertise – Client satisfaction	– Other businesses – Consulting firms
Mobile sensors	Sensor	Control gateway	– Reliability – Transmission rate	– Other sensors – Control gateways

Consider the first example of TV News. A viewer can generally only watch one news channel at a time (*i.e.*, the channel forms an exclusive service provider). Now say that the viewer wants to watch coverage of the US presidential election. The speed of news coverage and bias in opinions are dynamic features that vary depending on the news anchor and guests. Some viewers may prefer accuracy over speed of coverage. Similarly, some viewers may prefer unbiased coverage whereas others may consciously prefer an opinion biased towards one end of the political spectrum. Therefore, viewers distribute themselves over multiple channels and benefit from the choice of options. If a viewer wants updated information on the realized speed of coverage and the bias in opinions currently being expressed on channels not being watched, they have a few

[4]If one were to assume known *translation models* such that one agent's perspective can be translated into that of another agent in conjunction with communication of the observed values, then this criterion may be relaxed; however, we do not assume this generalization in our work.

options: (i) rapidly *channel-surf* amongst the options, (ii) stay in communication with family and friends who may be watching other channels, or (iii) gather information about the features of interest through social media (e.g, Twitter). The first option is akin to the EXP3 algorithm in its isolated exploration and exploitation. The second and third options are akin to Negotiated Learning in recognizing the value of obtaining information from other agents.

Formally, we can then state an instantiation of the *TV news channel selection* problem of a viewer agent as a Negotiable Entity Selection Process:

$$\langle \mathbf{K}, \mathcal{Z}, \varphi(\mathbf{S}, \mathcal{F}), \mathbf{N}, \mathcal{A}, \mathrm{T}, \mathrm{R} \rangle$$

where:

- The set of agents $\mathcal{I}$ contains (i) specific family members and friends with whom the viewer agent can communicate, and (ii) various sources of social media. Family and friends may form one or more agent classes with similar (c, τ, x) negotiation parameters and Twitter and Facebook may be in one or more other agent classes. For example, we map $\mathcal{I}$ to $\mathcal{K}$ = $\{$*FamilyViewer, FriendViewer, SocialMedium*$\}$ to form $\mathbf{K}$.

- The set of negotiable entities $\mathcal{Z}$ is, for example, defined as $\{$*ABC, CBS, CNN, FOX, NBC*$\}$. The entity features $\mathcal{F}$ are $\{$*CoverageSpeed, OpinionBias*$\}$. Thus, $\varphi(s(t), \mathcal{F})$ generates $5 \times 2 = 10$ transformed states.

- The negotiation model $\mathbf{N}$ contains (c, τ, x) negotiation parameters for the $|\varphi(s(t), \mathcal{F})| \times |\mathbf{K}| = 30$ edges of the bipartite graph. For example, negotiating with agents in the *FamilyViewer* or *FriendViewer* class may yield faster (low τ) and more accurate information (high x) but at a higher cost (high c). $\mathbf{N}$ may be given or learned.

- There are 10 negotiation actions in $\mathcal{A}_1(t)$ corresponding to the 10 states in $\varphi(s(t), \mathcal{F})$. The set of entity selection actions $\mathcal{A}_2(t)$ corresponds to the five channel entities in $\mathcal{Z}$.

- The viewer may have some understanding of the transition function T from published channel programming guides but the exact state transitions probabilities are unknown.

- The reward function R is a combination of the viewer's intrinsic preferences and the realized entity features on the different channels—it is likely to be unknown.

The other example problems in Table 5.1 follow similar reasoning and can be formulated by analogy. We believe they are self-explanatory, so in the interest of brevity, we do not explicitly

represent each of them as Negotiable Entity Selection Processes.[5] Note that while we only identify two entity features and two neighboring agent classes for each problem, more of each can certainly be defined as needed.

5.5 Chapter Summary

This chapter introduced models and algorithms for a decision-making agent faced with problems of sequential entity selection based on dynamic partially observable features in a semi-cooperative multiagent environment. The *Negotiated Entity Selection Process* (NESP) is a novel representation that captures *negotiable partial observability*—the semi-cooperative multiagent structure that is exploited by our ATTRACTION-BOUNDED-LEARNING algorithm. A key factor in the success of our approach is in recognizing the importance of separating, where feasible, the decision-making criteria for exploration and exploitation. The use of *Attractions* to separately capture metrics for negotiation (exploration) and entity selection (exploitation) is a critical element in the design of ATTRACTION-BOUNDED-LEARNING. The use of a bipartite graph structure to represent the *negotiation model* in an NESP is a flexible and powerful mechanism to capture extensive information about the domain and the multiagent environment while also remaining simple enough to be learned when it is unknown. Moreover, the use of the *state transform function* enables us to define abstract metrics for entity selection while mapping those entities to well-defined *entity features* grounded in the decision-making agent's environment.

[5] Our previous work on mobile sensor networks in DARPA's *LANdroids* domain provides more context, if needed, for the last example [Reddy and Veloso, 2011b].

Chapter 6

Learning Customer Agents

To validate the applicability of Negotiated Learning in the development of learning Smart Grid customer agents, this chapter presents experimental analysis for the two case studies that we have formulated in Chapter 5: *variable rate tariff selection* and *capacity aggregate management*.

6.1 Setup and Primary Results

In this section, we describe the experimental setup and present the primary results for both example problems. Then, in Section 6.2, we further analyze sensitivity, scaling and self-play properties within the context of the capacity aggregate management problem.

6.1.1 Variable Rate Tariff Selection Experiments

Recall from Section 5.2.3 that we assume the existence of sets of imputation methods and forecasting methods. Specifically, we have defined:

Imputation Methods $= \{\text{GlobalMean, CarryForward, BackPropagation, Interpolation}\}$

Forecasting Methods $= \{\text{Lag24, AR}(1)\text{, ARMA}(1,1)\text{, SARMA}(1,1) \times (1,1)_{24}\}$

6.1.1.1 Experimental Setup

We define a set of **agent configurations**:

- **Baseline**: An agent that explores the available tariffs using the available imputation methods and forecasting methods but does not negotiate for information.

- **Informed**: An agent that is fully informed about the price histories of *all* tariffs and uses those, instead of negotiations, to compute Attractions.[1]
- **MinRegret**: An agent that uses one of the algorithms from the EXP3 family to guide its exploration-exploitation behavior.
- **NLModelFree**: An agent that uses ATTRACTION-BOUNDED-LEARNING but does not know the agent classification map in $\mathbf{K}$ nor the negotiation model **N**, and consequently chooses negotiation partners *randomly*.
- **NLModelKnown**: An agent that uses ATTRACTION-BOUNDED-LEARNING and is fully informed about the agent classification map in **K** and the negotiation model **N** including the (c, τ, x) parameters for each negotiation action.
- **NLModelLearn**: An agent that does not initially know the agent classification map in **K** nor the (c, τ, x) parameters in the negotiation model **N**, but learns them using ATTRACTION-BOUNDED-LEARNING over successive episodes.

We simulate agents of the various configurations using the Power TAC simulation platform [Ketter et al., 2013] that we introduced in Chapter 3. We simulate 10 agents for each of the 6 agent configurations above and report the average results by agent configuration. We arbitrarily choose, without loss of generality, episodes of 240 time steps, which corresponds to 10 days with an hourly metering period. We generate consumption capacity forecasts using noise-added subsets of real hourly consumption data for homes in Southern California [San Diego Gas & Electric, 2012].

We use a combination of heuristic and reference simulation data to generate tariff prices. The top subfigure of Figure 6.1 shows hourly prices over 10 days for 3 tariffs:

- A fixed default utility tariff T1 (dotted black line).
- A stable dynamic TOU tariff T2 (dashed blue line) where each cycle of the pattern represents one day and prices are generally higher 8am-8pm.
- Another non-stationary tariff T3 (solid red line) whose prices are initially more volatile and higher than average compared to those of T2; however, over time the range and average of the prices decreases so that it becomes a more attractive option than T2 approximately half-way through the episode.

[1] Note that the Informed agent only knows the historical prices and not the future prices of the tariffs, so it still relies on forecasting to compute Attractions.

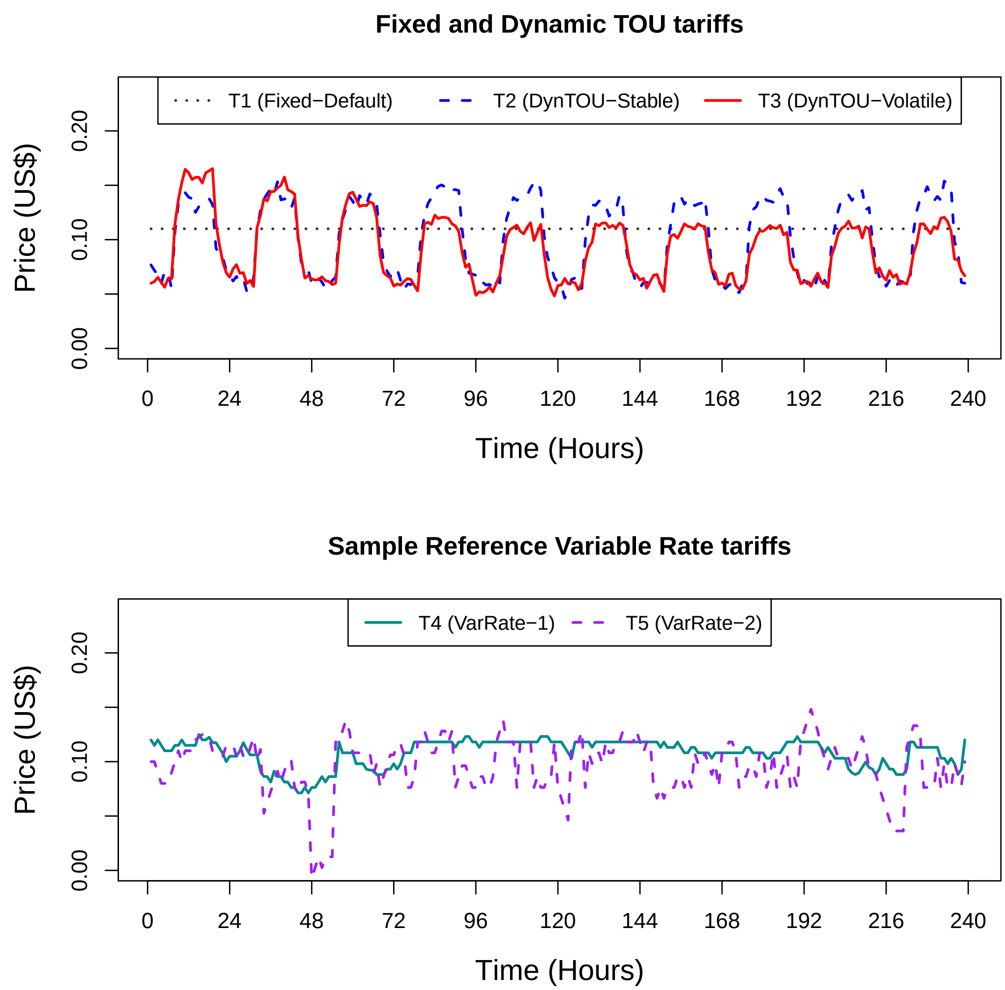


Figure 6.1: Heuristically generated fixed and dynamic TOU prices (top), and sample reference variable prices (bottom) used in the variable rate tariff selection experiments.

The bottom subfigure of Figure 6.1 is an illustration of two additional tariffs, T4 and T5, that are drawn independently for each simulation episode from a reference set of tariff prices offered by simulated competitive electricity brokers. The brokers employ various pricing strategies including variable rates indexed to a wholesale electricity market [IESO, 2011] and related adaptive and reinforcement learning-based pricing strategies that optimize for the broker's profit maximizing goals [Reddy and Veloso, 2011c].

The remainder of this section presents the primary results obtained using the above experimental setup. We first describe findings on **cost savings**, *i.e.*, reductions in cumulative costs as defined in Equation 5.2, for various agent configurations. We then delve into:

- the evolution of Attractions,
- the impact of the semi-cooperative assumption, and
- the outputs from the imputation and forecasting methods.

6.1.1.2 Primary Results

Figure 6.2 demonstrates the value of exploiting the multiagent structure of the variable rate tariff selection problem. The y-axis shows the cost savings for various agent configurations relative to the Baseline agent (dotted black line). The Informed agent (solid purple line) demonstrates the significant opportunity for cost savings when the agent has full information about the dynamic prices for all tariffs. MinRegret agents using EXP3.P (dashed brown line) show negative savings, *i.e.*, higher cost than for Baseline.[2] Recall that each line in the plot represents averaged results for 10 agents of that configuration.

The fully informed agent sets the upper bound on cost savings, but it is unrealistic in our semi-cooperative setting. Figure 6.3 shows approximate bounds for agents that acquire information through negotiation. The flat line represents the same Baseline as in Figure 6.2. An upper bound on cost savings for a Negotiated Learning agent is established by NLModelKnown agents, which are given an accurate negotiation model (solid green line). Conversely, the NLModelFree agents (dashed blue line) demonstrate the negative savings if a negotiation model is not used, *i.e.*, the agent chooses a neighbor to negotiate with randomly. The gap between the two new lines illustrates the value of the negotiation model.

Agents in different agent classes, $\mathcal{K} = \{$*Desirable, Undesirable*$\}$, are configured to exhibit different (c, τ, x) attributes, *i.e.*, charge different prices, require varying time periods to respond, and vary in reliability. Figure 6.4 shows NLModelBuild agents (dashed magenta line), which do not retain information learned about the negotiation model from one episode to another; their performance is comparable to NLModelFree agents. NLModelLearn agents (solid red line) improve upon that performance significantly by retaining learned negotiation models from one episode to another, with each episode using a possibly different set of tariff prices from the reference set and different real data subsets for the capacity forecasts. After several learning episodes, the performance of NLModelLearn agents approaches the performance of NLModelKnown agents.

[2] The performance of MinRegret agents using EXP3 or EXP3.S is similar to that of agents using EXP3.P.

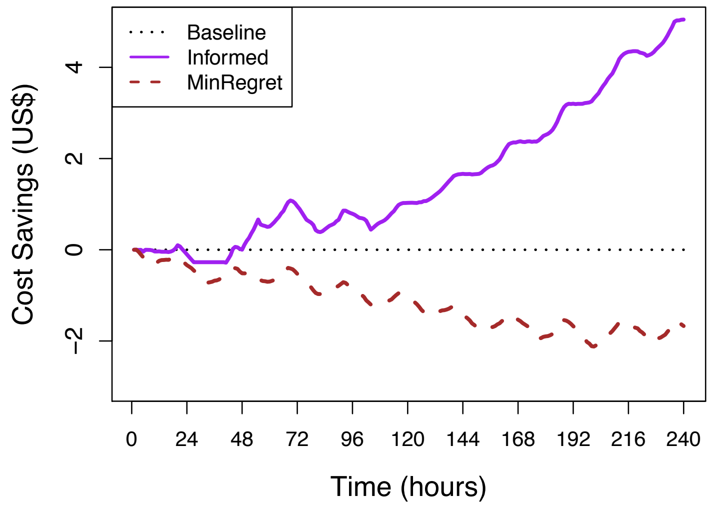


Figure 6.2: Cumulative cost savings over one episode for Baseline, Informed and MinRegret agent configurations.

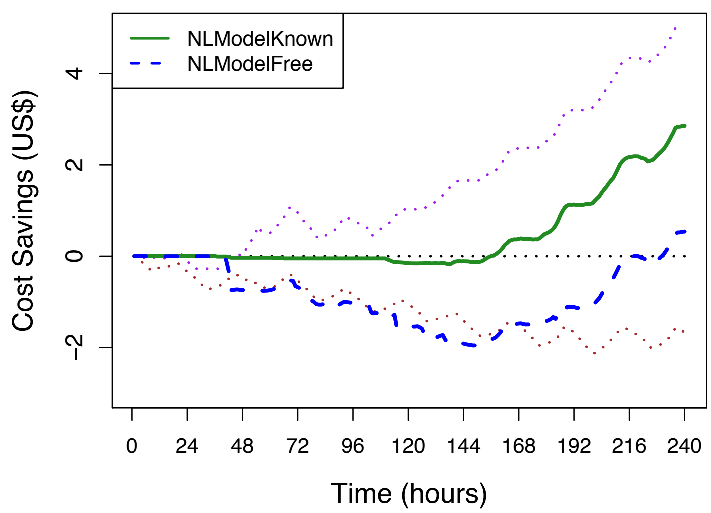


Figure 6.3: Cumulative cost savings over one episode for NLModelKnown and NLModelFree agent configurations.

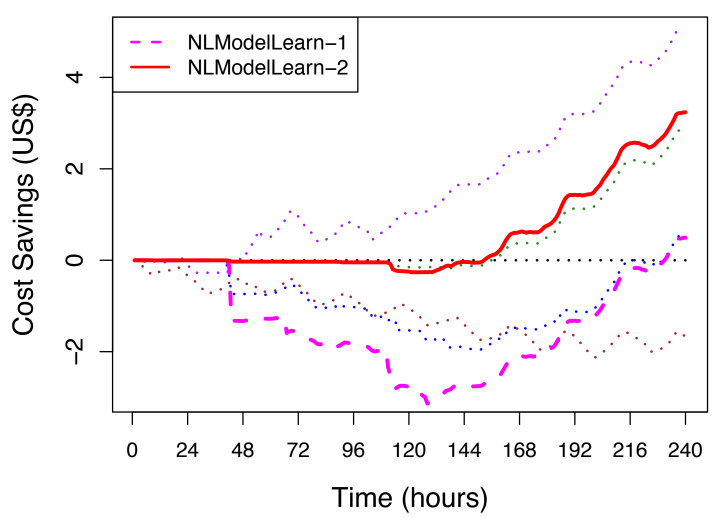


Figure 6.4: Cumulative cost savings over one episode for NLModelBuild and NLModelLearn agent configurations.

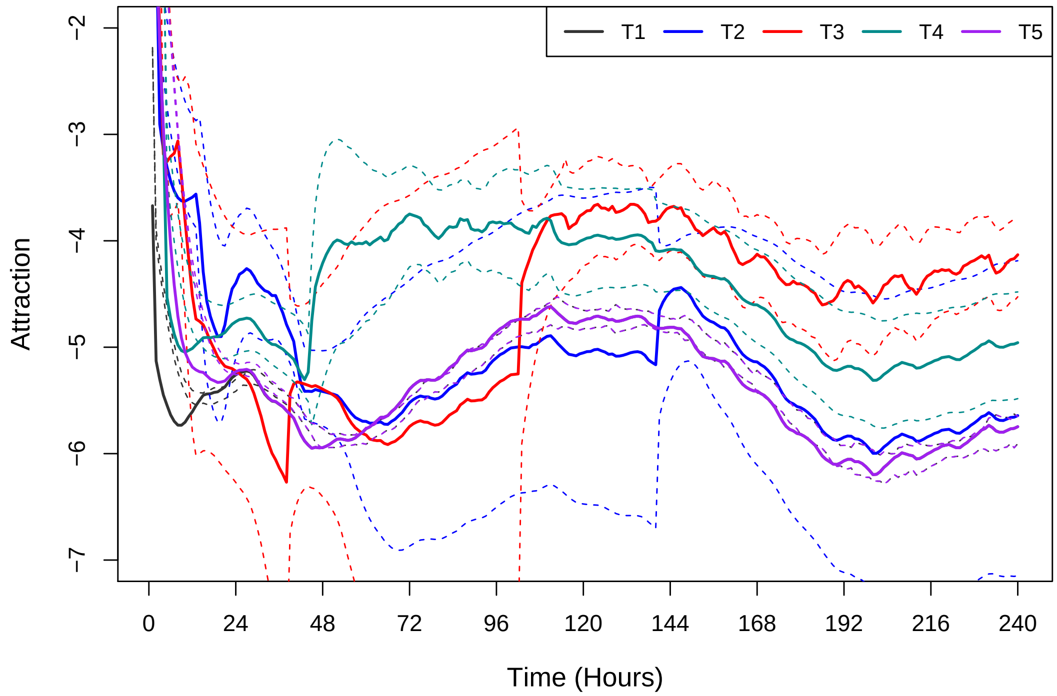


Figure 6.5: Evolution of the Attraction triple (μ, β^+, β^-) for each of the tariffs shown in Figure 6.1 over one episode for a typical Negotiated Learning agent.

6.1.1.3 Evolution of Tariff Attractions

To gain a deeper understanding of how the ATTRACTION-BOUNDED-LEARNING algorithm extracts the value of negotiated information, we now present an analysis of the Attractions underlying the behavior of the Negotiated Learning agents in the current example.

Figure 6.5 shows the evolution of the Attraction triple, (μ, β^+, β^-), for each of the tariffs presented in Figure 6.1 for a typical Negotiated Learning agent over one episode. The means, μ, for each Attraction are plotted using solid lines (their colors correspond to the colors of the tariffs in Figure 6.1). The corresponding upper and lower bounds, β^+ and β^-, for each Attraction are plotted as color-matched dashed lines. We note several interesting aspects in the figure:

- The Attractions are *not* normalized, so their evolution mirrors the absolute values of tariff prices and the ups and downs of the capacity forecasts, all else being equal. Consequently, if capacity is low at a particular time t, the Attractions are closer together at that t. Since the Attractions in this example represent average charges over the tariff evaluation horizon, H, they are nominally measured in US$ here. So, it's possible to compare the means, or bounds, of two Attractions using meaningful absolute values and not just percentages when

making exploitation, or exploration, decisions. For example, when evaluating whether to switch from tariff x to tariff x', the passing condition may be $\mu_{x'} > \mu_x + \$0.5$ where $\$0.5$ forms the *benefit threshold*, ξ, which is used to control how sticky the decision process is to its current selection.

- The Attraction update weight parameters ω_b and ω_e control the smoothness of the evolution. A relatively low *belief update weight*, ω_b, ensures that Attractions for tariffs about which the agent has no new information do not fluctuate wildly. Conversely, a higher *experience update weight*, ω_e, ensures that the agent recognizes and acts upon new information quickly. In Figure 6.5, $\omega_b = 0.2$ and $\omega_e = 0.6$. In Section 6.2, we illustrate the sensitivity to these parameters in the context of the capacity aggregate management problem.

- Referring to the tariff prices in Figure 6.1, we anticipate that the volatile dynamic TOU tariff, T3, would have higher Attraction (lower average charges) than the stable dynamic TOU tariff, T2, approximately 4 days into the episode. We then hypothesize that a Negotiated Learning agent would exploit such an opportunity. Indeed, we see in Figure 6.5 that at $t \approx 100$, the μ of the Attraction for T3, which had thus far been lower than that for T2, jumps up and over the μ for T2. We attribute this change to information that the agent obtained through a negotiation that was initiated a few time steps earlier. Indeed again, studying the upper bounds, β^+, for T3 and T4, we see that around $t \approx 72$, the β^+ for T3 overtakes that for T4, thus triggering a negotiation. This sequence of actions validates the intended role of Attractions in the ATTRACTION-BOUNDED-LEARNING algorithm.

While the last bullet point above bears out the design of the experiment, interestingly, both of the randomly drawn tariffs, T4 and T5, add interesting twists to the narrative.

- First, T4 in this particular experiment is an attractive tariff compared to T2 for much of the episode because it's prices do not exhibit the TOU peaks like those in T2's prices (see Figure 6.1). The Attraction mean values for the two tariffs reflect such a favorable comparison starting at $t \approx 40$ and the agent switches from T2 to T4 at that time. This demonstrates that ATTRACTION-BOUNDED-LEARNING is able to respond to unexpected opportunities that were not explicitly designed into the experiments.

- Second, we see from Figure 6.1 that T5 may also be attractive for some periods because its prices dip down significantly at certain times, *e.g.*, $t = 48$. However, its volatility makes it difficult to judge how it compares to the other tariffs, especially T4 and T3. This scenario provides an opportunity to answer a critical question that we address next.

6.1.1.4 Impact of the Semi-Cooperative Assumption

How do we explain the difference in performance (cost savings) between the fully informed agents in Figure 6.2 versus the approximate upper bound on Negotiated Learning agents in Figure 6.3?

To answer this question, we refer to Figure 6.6, which traces the Attractions as computed by an Informed agent configuration, *i.e.*, an agent that uses full knowledge of price histories for each tariff along with the forecasting methods used by the Negotiated Learning agents. It is worth emphasizing that since the agent has entire price histories, it has no *imputation uncertainty* as defined in Section 5.2.1 and therefore does not need any imputation methods.

We find some key differences in the evolution of the Attraction means in Figure 6.6 versus those in Figure 6.5. Most importantly, the highly volatile variable rate tariff T5 has the highest Attraction for much of the episode. The Informed agent is able to use this information to successfully navigate the volatility of T5 to obtain the best prices. Secondarily, also note that the Attraction for T4 is not as high as in Figure 6.5. Given its fully populated tariff price histories, the Informed agent is able to forecast the average customer charges more accurately than the Negotiated Learning agents.

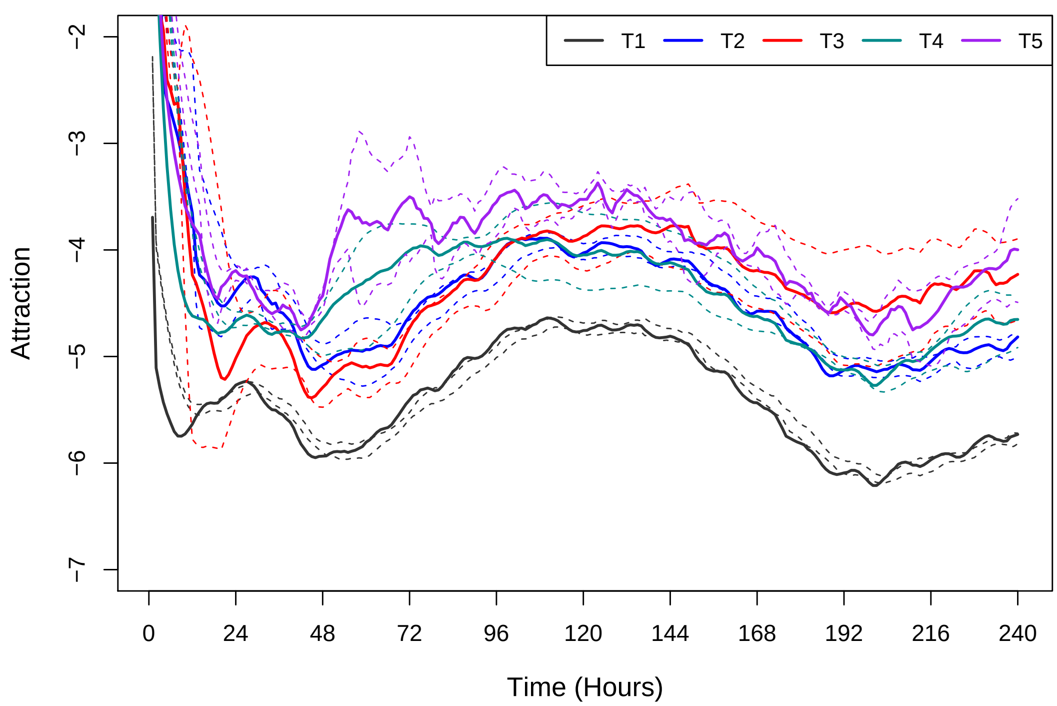


Figure 6.6: Evolution of the Attraction triple (μ, β^+, β^-) for each of the tariffs shown in Figure 6.1over one episode for an Informed agent configuration.

Moreover, since the price histories are updated at every time step, the Informed agent is able to update its price forecasts using the experience weight, ω_e, at every time step—this ensures that it is very responsive to price movements. Furthermore, since it updates its forecasts at each time step, it only relies on its forecasts being accurate for one time step, thus reducing its reliance on the forecasting methods.

On the other hand, referring back to Figure 6.5, we note that the Negotiated Learning agent recognized the high Attraction of T4 quickly but could not do the same for T5 because of T5's irregular evolution, *i.e.*, low 24-hour correlations and low autoregressive coefficients. Coincidentally, the high Attraction of T4 also sets a higher bar for the bounds of T5 to overtake before the agent will try to negotiate for information about T5 again.

Overall, this causes the Negotiated Learning agent to miss the *fleeting* or *spiky* opportunities presented by T5, which account for the performance difference between the Informed agents and the best Negotiated Learning agent configurations.

While this scenario identifies an essential limitation of Negotiated Learning techniques, and explains the performance gap compared to fully informed agents, it also offers proof that this is a necessary artifact of our *semi-cooperative* assumption. Indeed, if we assume *zero* negotiation costs between agents, each Negotiated Learning agent would acquire all price histories available from all other agents.

If the agents' capacity profiles are such that they sufficiently distribute their subscriptions over all tariffs, *i.e.*, each tariff has at least one subscribing agent, then each Negotiated Learning agent would *match* the performance of the Informed agent. This means that *the performance of a group of* ***semi-cooperative*** *Negotiated Learning agents with zero negotiation costs converges to the performance of similar agents in a fully* ***cooperative*** *environment.*

More realistically, if we assume that some of the tariffs would not acquire any subscribers initially, it is possible that in an environment consisting only of Negotiated Learning agents (*i.e.*, self-play), those tariffs will remain unexplored given high exploration costs, thus opening up the possibility of missed opportunities. So, it would be rational for the agents to cooperatively explore the unsubscribed tariffs; however, that contradicts our semi-cooperative assumption, thus necessitating the performance gap that we observe.

6.1.1.5 Evolution of Tariff Price Forecasts

Imputation methods and *forecasting methods* play an important role in determining tariff Attractions, which in turn control the negotiation (exploration) and tariff selection (exploitation) behaviors of the Negotiated Learning agents. Recall that the dot product of each tariff price fore-

cast with each capacity forecast over a lookahead window yields one value that represents the sum of the expected charges for that tariff-capacity-lookahead combination. Each sum charge is divided by the corresponding lookahead window size to yield the set of *average charge* values. The mean and standard deviation of this set are used as the means, μ, and to compute the bounds, $\beta^{\pm}$, of the tariff Attractions.

We noted earlier that the imputation and forecasting methods are domain-specific in that they assume some prior knowledge of how tariff prices, or more generally *entity features*, typically evolve. However, there are some general characteristics to look for when determining the set of models to use for any given problem. Abstractly, we would like the forecasts generated by the models to sufficiently disperse so that the Attraction bounds are not restrictive and also sufficiently converge so that the mean is a valid metric for the expected value of the distribution of average charges.

We use Figures 5.8-5.10 to consider the *accuracy* and *robustness* of the price forecasts generated using the imputation and forecasting methods identified in Section 5.2.3 for the variable rate tariff selection case study. We have 16 price forecasts from the 4 imputation methods and 4 forecasting methods in this experimental setup. Figure 6.7 shows 3 of the 16 forecasts for tariff T3. The black line represents the realized tariff prices for a particular episode. The rainbow-colored lines are a series of 240 lines, one for each time step, each of length equal to the tariff evaluation horizon, $H = 24$. The lines are colored progressively such that the initial forecasts are red and the forecasts at the end of the episode are blue.

The forecasts in Figure 6.7 are obtained from a fully informed agent, so the imputation methods are irrelevant as we can see from the identical top and middle subfigures. Note further that the Lag24 forecasting method becomes fairly *accurate* over the second half of the episode; this is not surprising since the prices for T3 are generated heuristically as a function of the day-earlier price. On the other hand, we see that the bottom subfigure exhibits much weaker correlation with the realized prices. This is attributable to the auto-regressive and moving average components of the SARMA model which bias the forecasts towards the global mean of the observed series. Comparing the Lag24 and SARMA model forecasts here, we might be tempted to discard the SARMA model.

However, we see in Figure 6.8 that SARMA is more *robust* than Lag24 and therefore worth including in the set of models. Note that these forecasts are for tariff T4, also from a fully informed agent. The price pattern of T4 was originally generated by an adaptive or learning broker that exhibits intermittent patterns of lag-24 correlations but is irregular on the whole. The Lag24 model's stronger assumptions lead to misguided forecasts as seen in the 24-hour cycles around $t = 100$. On the other hand, the SARMA model forecasts more conservatively during

the same period. While its forecast is not accurate either, it is less wrong than Lag24 and adds a degree of *robustness* to the derived Attraction.

Finally, in Figure 6.9 we can observe the effects of the imputation methods because these forecasts for T3 are obtained from a Negotiated Learning agent with missing information during various periods. Note the discontinuities in the forecasts from $t \approx 80$ to $t \approx 150$ (compare to Figure 6.7). During that period, the agent seems to have obtained new price samples or price histories that steadily improved the forecasts. In fact, we know from the corresponding evolution of Attractions that the agent obtained negotiated information about T3 at $t \approx 80$ and switched to it at $t \approx 150$. Note how the GlobalMean imputation method generates spiky forecasts whereas the Interpolation model generates angular forecasts; one may not necessarily be better than the other but together they add dispersion to the set of forecasted average charges and *robustness* to the derived Attraction.

6.1.2 Capacity Aggregate Management Experiments

The remainder of the experiments in this section demonstrate how our ATTRACTION-BOUNDED-LEARNING-based solution to the capacity aggregate management problem generates cost savings for customers. These savings are generated by minimizing usage charges while controlling the shifting penalties, switching costs, and negotiation costs, which are all included in the reward function $\mathbb{R}$, as specified by Equation 5.13.

In these experiments, we construct the minimal scenario that sufficiently represents the problem and provides a tractable platform for subsequent experiments where we analyze the sensitivity to various environmental and algorithmic tuning parameters. In Section 6.2, we scale the scenario to larger numbers of agents and find that the primary results obtained here are valid for larger scenarios.

We simulate agents in 20-day episodes with hourly metering, using the Power TAC simulation platform. We generate capacity profiles for the agents using noise-added subsets of real hourly consumption data for customers in Southern California [San Diego Gas & Electric, 2012].

The minimal scenario includes three agents of the following classes:

1. **Stable Component**: A *component* agent of this class contributes capacity to the *aggregate*. The hourly capacity follows a 24-hour profile in the shape of the top-left subfigure of Figure 6.10. Gaussian noise is added for each iteration of the profile.

2. **Volatile Component**: A *component* agent of this class appears identical to a Stable Component agent early in a simulation episode, but over time exhibits a *drift* in its profile

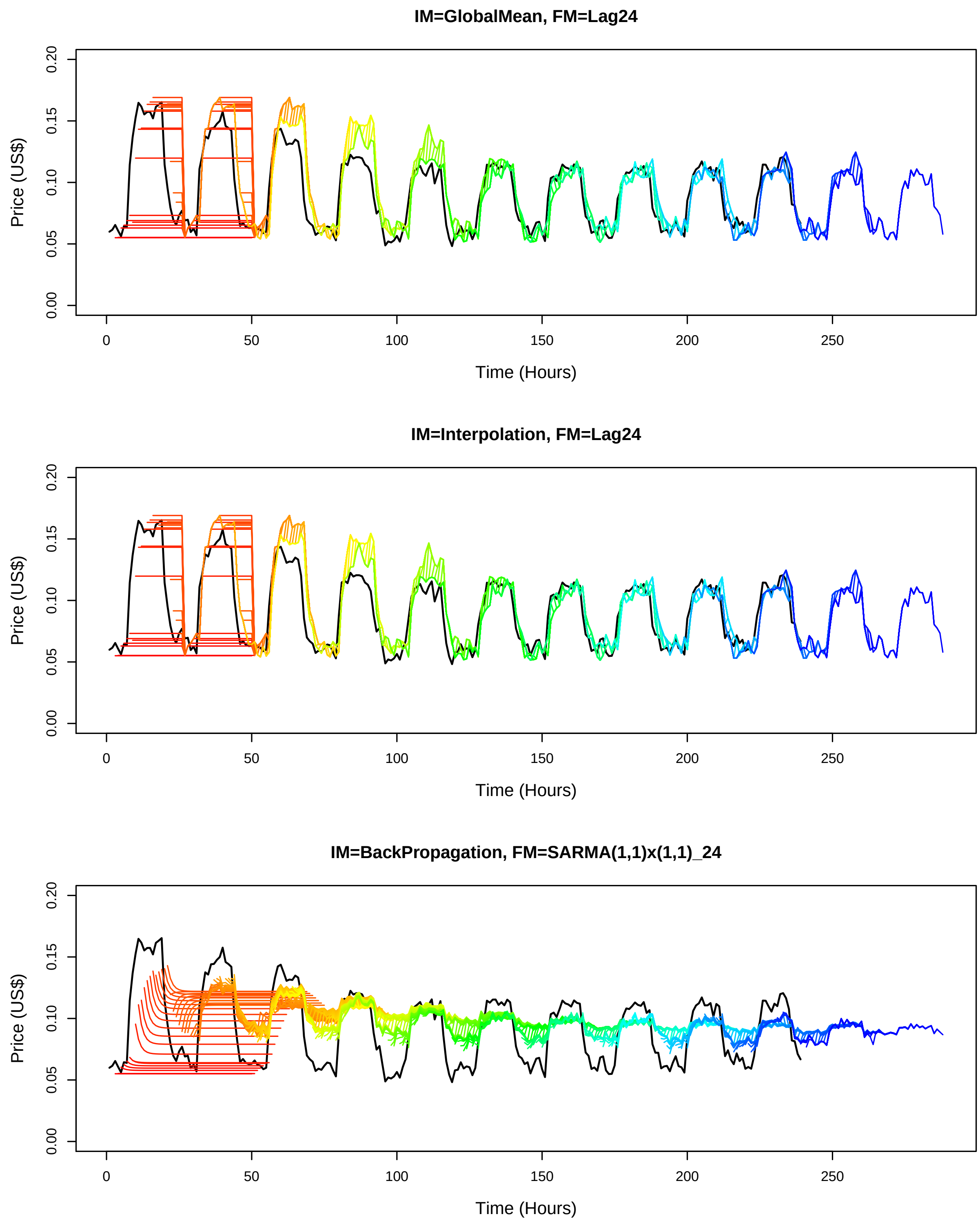


Figure 6.7: Each subfigure shows a series of forecasts for tariff T3 generated by an Informed agent using a sample combination of imputation methods (IM) and forecasting methods (FM).

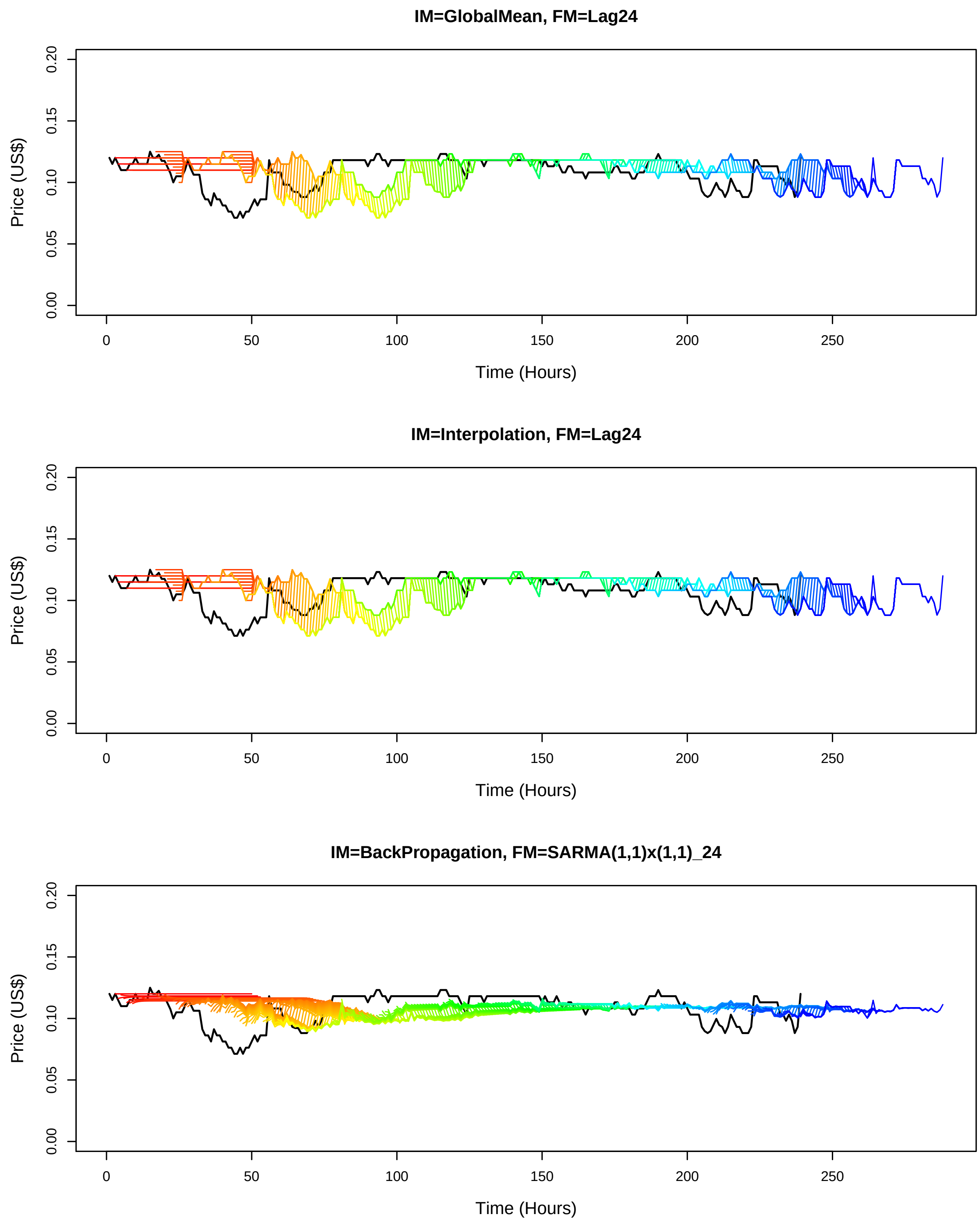


Figure 6.8: Each subfigure shows a series of forecasts for tariff T4 generated by an Informed agent using a sample combination of imputation methods (IM) and forecasting methods (FM).

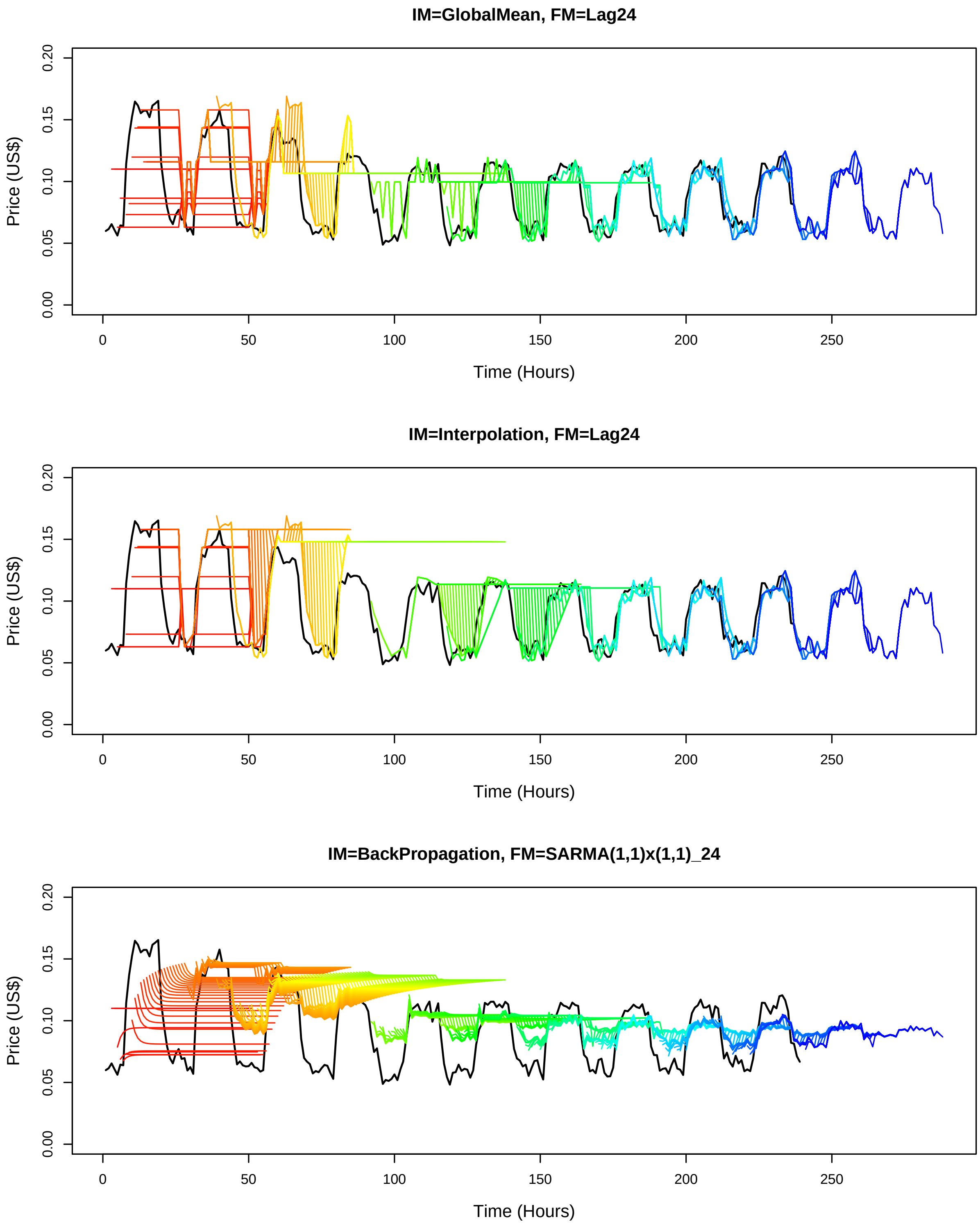


Figure 6.9: Each subfigure shows a series of forecasts for tariff T3 generated by a Negotiated Learning agent using a sample combination of imputation methods (IM) and forecasting methods (FM).

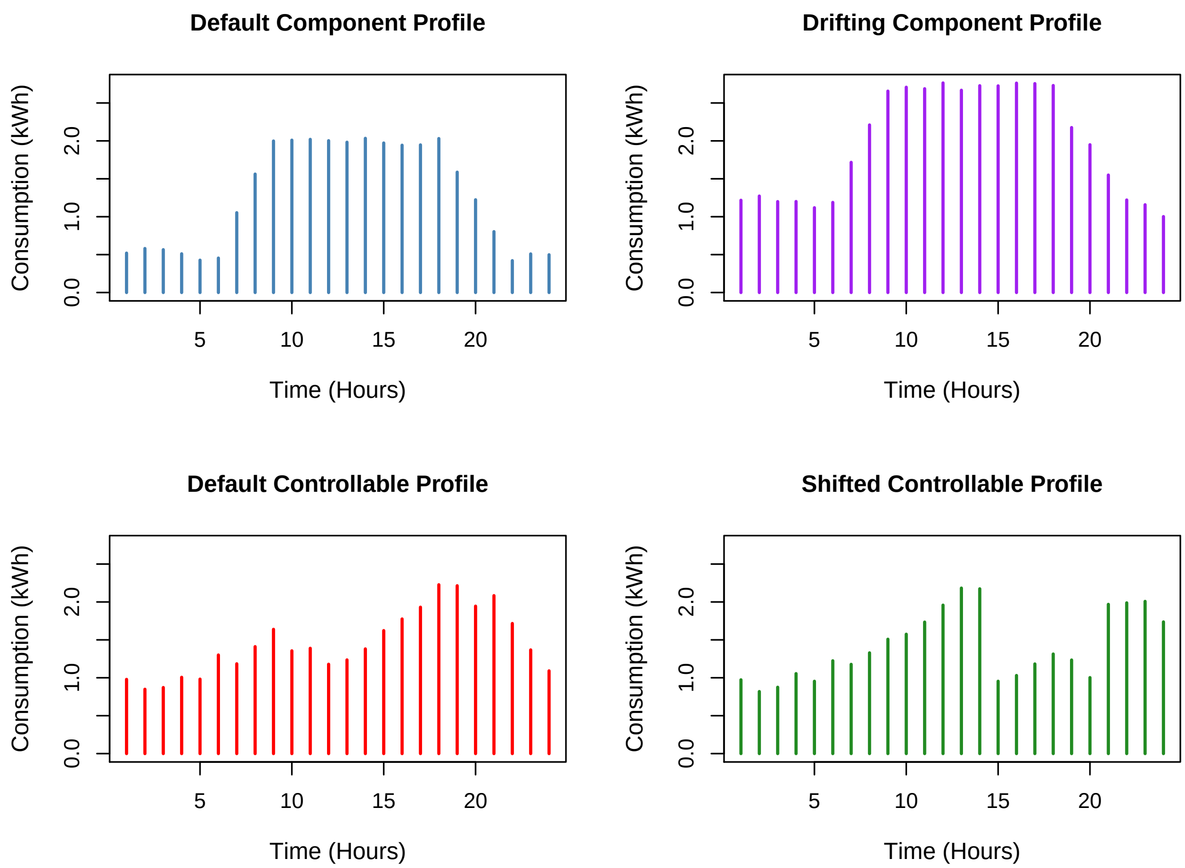


Figure 6.10: 24-hour capacity profiles for the Stable component, drifted Volatile component, and the default and shifted profiles of the Aggregator's controllable capacity.

towards a higher capacity contribution. The peak of the contribution is similar to the top-right subfigure of Figure 6.10. The effect of the drift can be seen in the top-right subfigure of Figure 6.11, which shows the contributed capacity over one full episode. Note that the drift is reversed over the second half of the episode.

3. **Aggregator**: The singleton agent in this class contributes a *controllable capacity* to the aggregate and is also responsible for managing the usage charges associated with the aggregate. The controllable capacity is drawn from one of two profiles: (i) the *default* preferred profile shown in the bottom-left subfigure of Figure 6.10, or (ii) the *shifted* profile shown in the bottom-right subfigure. We assume here that there is only one shifted profile, but we later describe how scaling to many shifted profiles does not qualitatively affect our results.

We further assume that the applicable tariff is a tiered dynamic time-of-use (TOU) tariff, as described in Section 5.3, that designates peak hours as 2pm to 8pm. In the minimal scenario, the

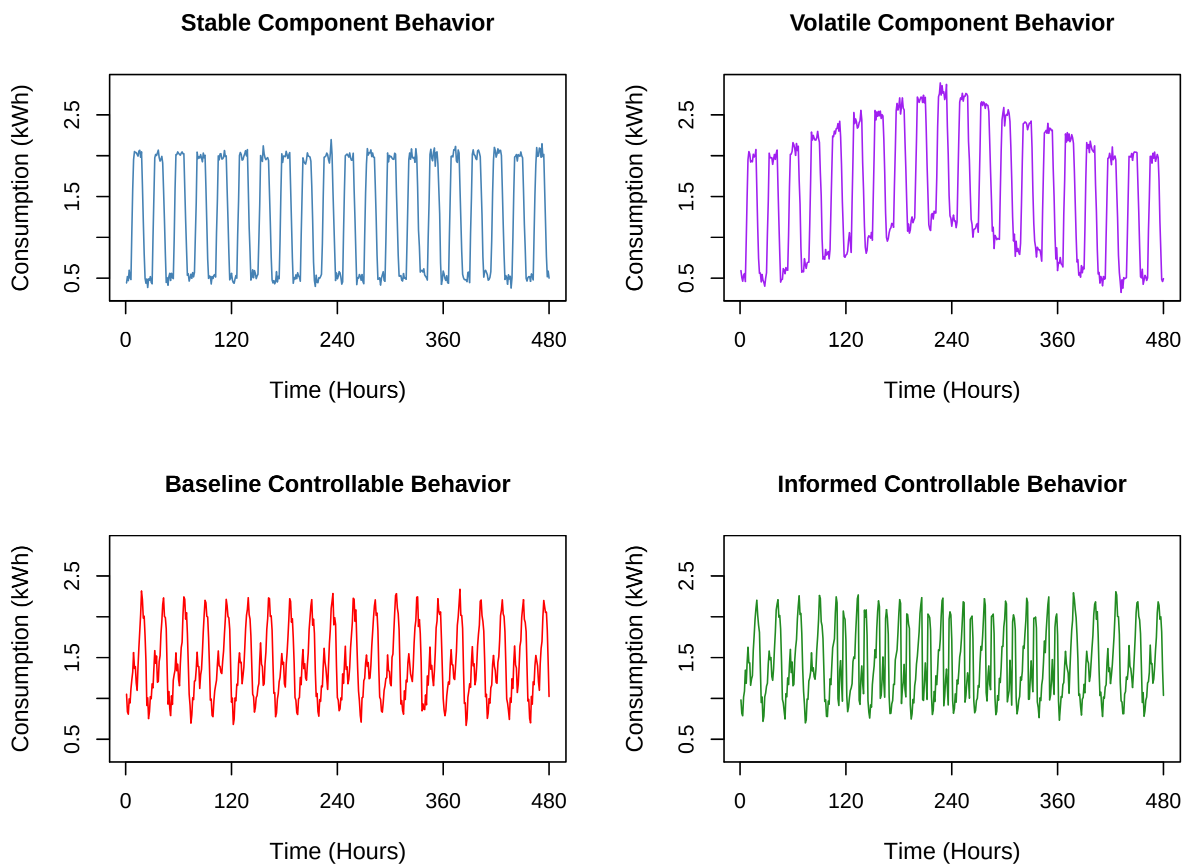


Figure 6.11: Capacity profiles adopted by Stable and Volatile components over an episode, and the *Baseline* and *Informed* behaviors for the Aggregator's controllable capacity.

tier threshold at which higher rates apply during the peak period is set to 6.5 kWh—the threshold is scaled proportionally in later experiments where the scenarios have more agents.

The basic experiment is designed such that under the assumed tariff structure, given the stable and volatile capacity contributions of the two aggregate components and the default profile of the controllable capacity, the capacity aggregate surpasses the tariff's tier threshold for the middle half of an episode, approximately $120 < t < 360$. During that period, the Aggregator agent would benefit from adopting its shifted profile.

In our experiments, we label this as the **Informed** behavior and it's shown in the bottom-right subfigure of Figure 6.11. The non-shifting behavior shown in the bottom-left subfigure is labeled the **Baseline**. We define two additional behaviors for the Aggregator agent:

- **Negotiated Learning**: The profile selection decision process is represented as the NESP described in Section 5.3, with no knowledge of the agent classification map in $\mathbf{K}$ nor the

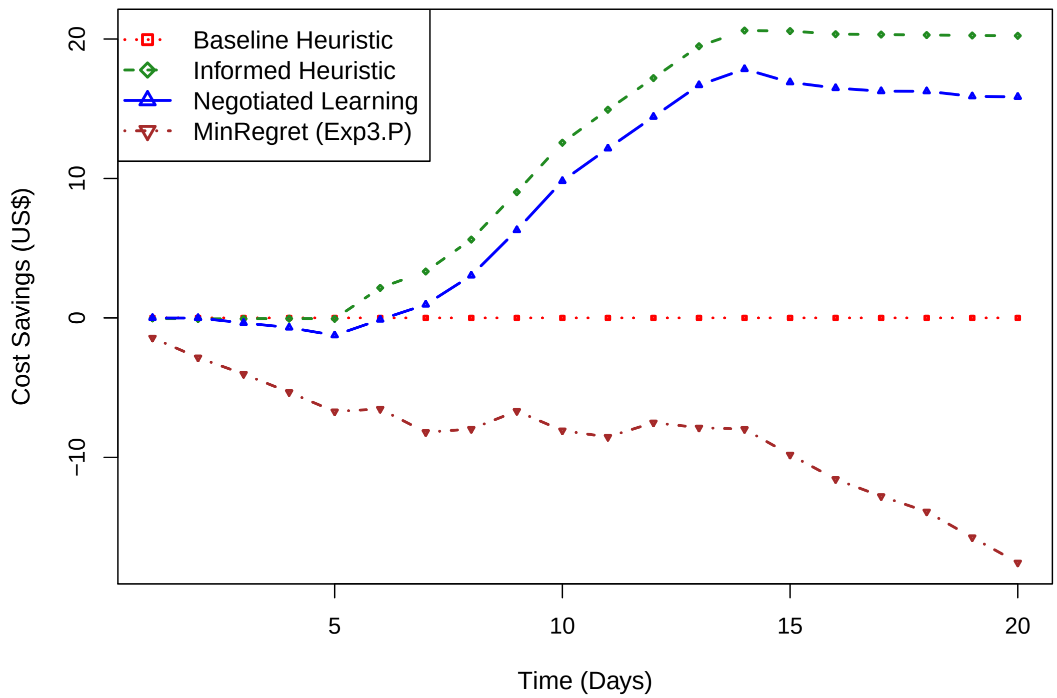


Figure 6.12: Cumulative cost savings over one episode in the minimal capacity aggregate management scenario for the Baseline, Informed, MinRegret, and Negotiated Learning behaviors.

(c, τ, x) edge parameters of the negotiation model $\mathbf{N}$, and the ATTRACTION-BOUNDED-LEARNING algorithm is applied to obtain the selection policy.

- **MinRegret**: The profile selection decision process is represented as in Equation 5.13 and EXP3.P is applied to obtain the selection policy.

We simulate each behavior of the Aggregator agent with the Stable and Volatile components to obtain the *cost savings* results shown in Figure 6.12. The savings are computed as the difference between the accrued costs for a particular behavior and the corresponding costs of the Baseline behavior, averaged over 10 iterations. The Informed heuristic sets the approximate upper bound for the cost savings if the Aggregator agent shifts its controllable capacity profile at the optimal times. The bound is approximate since the heuristic switches profiles exactly at $t = 120$ and $t = 360$ whereas the aggregate surpasses the tier threshold approximately in that period because of the noise in the capacity profiles.

The Negotiated Learning agent performs slightly worse than the Baseline initially as it pays the added costs of negotiation with the component agents but makes up for those costs by select-

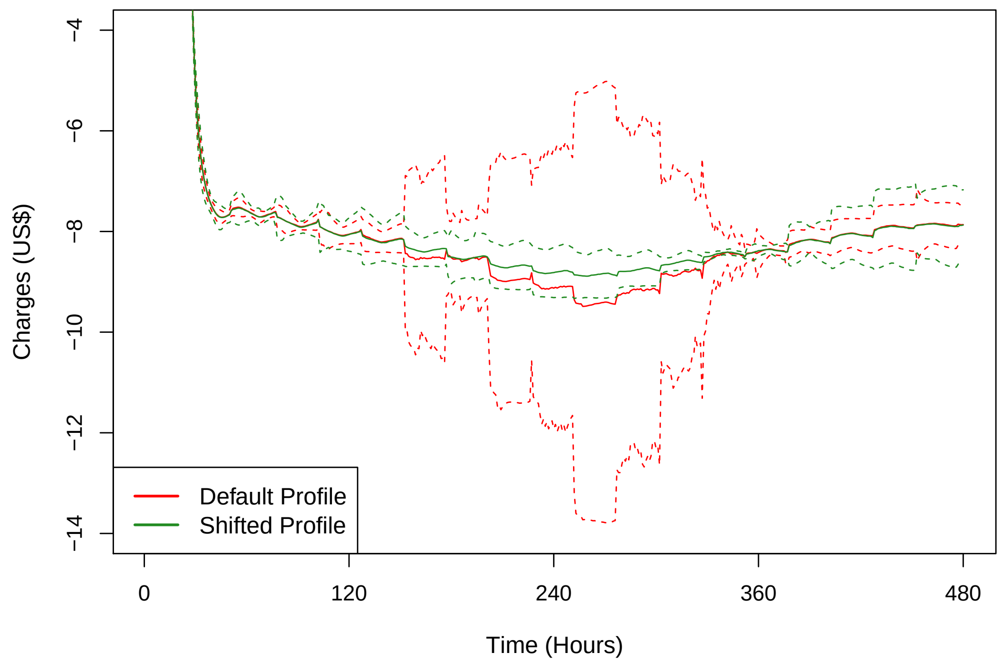


Figure 6.13: Evolution of Attractions, measured in average hourly charges, for a Negotiated Learning agent's *default* and *shifted* profiles.

ing the shifted profile from day 6. It gives up some of those relative gains towards the end of the episode as it evaluates switching back to the default profile.

To further understand the poor performance of the MinRegret behavior, it is useful to briefly delve behind the scenes of the ATTRACTION-BOUNDED-LEARNING and EXP3.P algorithms as they are applied in this context. Figure 6.13 shows the evolution of the Attractions for the default and shifted profiles of D's controllable capacity. As expected, the Attraction means are essentially identical initially, but over time they diverge significantly and then converge again. Those dynamics are reflected in Figure 6.14, which shows the number of times that an agent switches profiles within a single day. When the Attraction means are not sufficiently dispersed, the Negotiated Learning agent switches a few times per day, but as they disperse, the number of switches goes to zero.[3]

On the other hand, the number of switches by the MinRegret agent is significantly higher throughout the episode. Thus, the switching costs, c_s, are a major contributor to the worse per-

[3]With a tuned benefit threshold, $\xi > 0$, a Negotiated Learning agent can be optimized to further reduce the total number of switches in this experiment, bringing the number close to two.

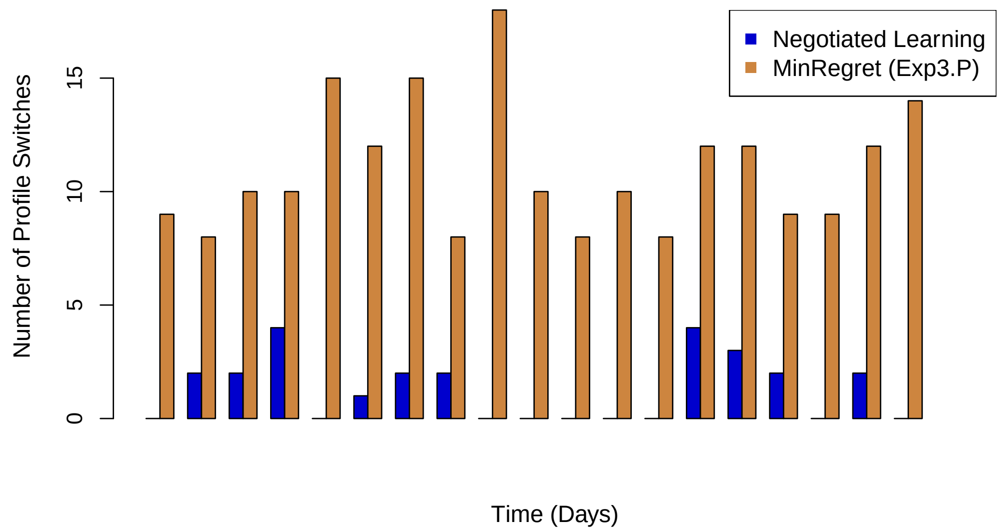


Figure 6.14: Comparison of the number of profile switches per day over an episode for a Negotiated Learning agent and a MinRegret agent.

formance of EXP3.P in this context. We continue the discussion of the impact of switching costs and other simulation parameters in the sensitivity analysis in the next section. We conclude our analysis of the primary experimental results by noting that our results do not represent that ATTRACTION-BOUNDED-LEARNING outperforms EXP3.P in all problem settings, but only in specific scenarios where the non-stationary multiagent structure of the problem and the switching costs are such that a separation of concern for exploration and exploitation is feasible and beneficial.

6.2 Sensitivity and Scalability

In this section, we analyze sensitivity to several simulation environment settings and explore the impact of scaling the number of agents, including a brief study of the emergent outcome of multiple agents using Negotiated Learning simultaneously in self-play.

6.2.1 Sensitivity Experiments

We employ a variant of the *capacity aggregate management* experimental setup–without added Gaussian noise in the capacity profiles–to study the sensitivity of performance of an Aggregator

agent that uses Negotiated Learning. The subfigures of Figures 6.15-6.17 trace the percentage cost savings in response to increases in the cost parameters of the simulated scenario and tuning parameters in the ATTRACTION-BOUNDED-LEARNING algorithm:

- **Switching cost,** c_s: We noted earlier in the context of Figure 6.14 that a typical Negotiated Learning agent switches between profiles far fewer times than a MinRegret agent. In the current noiseless scenario, the agent typically only switches twice, first when it notices the upward drift of the Volatile component as the Attraction means for the two profiles diverge and second when those means converge. So, the Negotiated Learning is practically insensitive to the switching cost in this noiseless setting, and mildly negatively correlated in the original noisy setting. In comparison, the MinRegret agent is significantly negatively correlated to changes in the switching cost.

- **Shifting penalty,** c_ρ: Since the reward function for the Aggregator agents has to balance the shifting penalty versus potential savings in usage charges from adopting the shifted profile, we intuitively expect that at a sufficiently high c_ρ, the Negotiated Learning agent will decide to forgo shifting and therefore converge with the performance of the Baseline agent. The MinRegret agent also shows a gentle trend towards the Baseline as c_ρ increases.

- **Negotiation costs**: As negotiation costs increase, the costs of obtaining negotiated information outrun the value of that obtained information in decreasing usage charges. So, if the costs are sufficiently high, the Negotiated Learning agent will again converge to Baseline behavior. Negotiation costs and the rest of the configuration parameters below only apply to Negotiated Learning agents, so we do not infer any patterns in the trend lines for the other agents.

- **Attraction bounds decay factor,** λ: It is possible in some situations that the known and imputed values for the entity features are stable enough that the forecasting methods mostly converge in their forecasts and therefore the bounds on that entity's Attraction are very narrow. If that's the case, that entity may not be explored again for a while, if ever. So, we force the bounds to slowly diverge over time to trigger exploration. We can see from the bottom subfigure of Figure 6.16 that the current example does indeed exhibit this situation at the beginning of the episode and therefore requires a non-zero value for λ. In certain problem settings, it may be beneficial to have λ as a function of some entity features. For example, consider a variant of the current scenario where different capacity aggregates consist of different Component agents; then the λ for that aggregate could depend on the ratio of Volatile and Stable components in the aggregate.

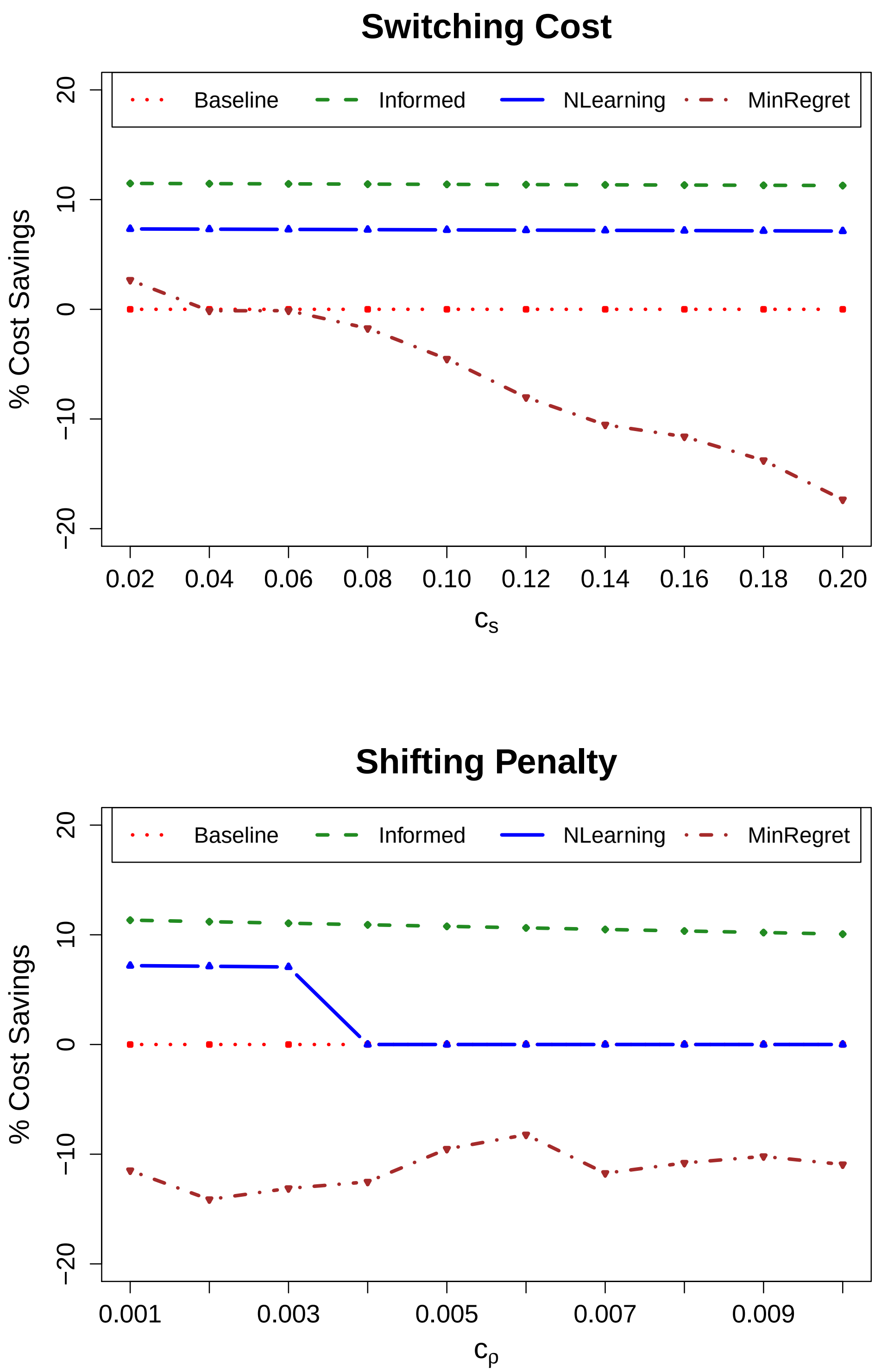
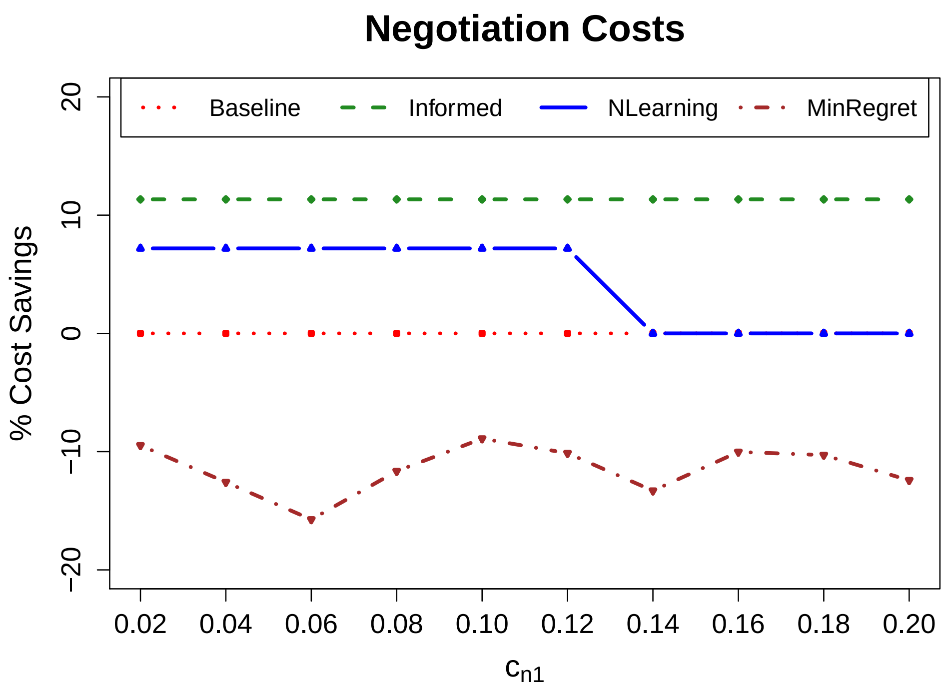


Figure 6.15: Sensitivity of percentage cost savings in response to increasing values of (a) switching cost c_s, and (b) shifting penalty c_ρ.



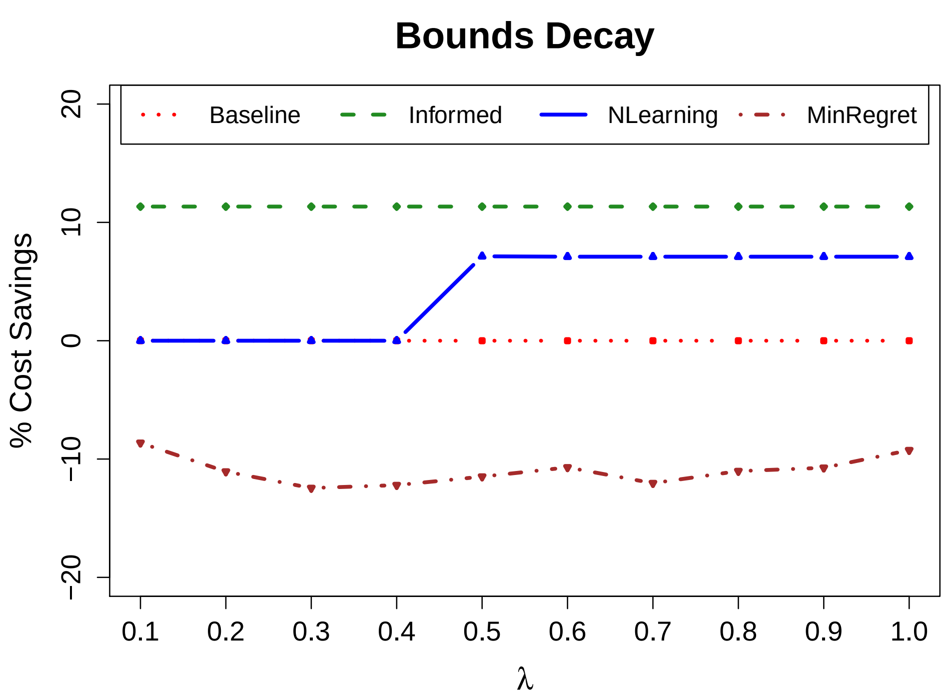


Figure 6.16: Sensitivity of percentage cost savings in response to increasing values of (a) negotiation costs, and (b) Attraction bounds decay factor λ.

- **Negotiation budget factor,** γ: The negotiation budget at a particular time step is computed as γ times the *potential benefit*—the difference in the Attraction bounds, β^+ or β^-, between the currently selected entity and an alternate entity.[4][5] In the current example, performance increases monotonically with γ, but in general is likely to be convex so that it can be optimized through training episodes.

  When the negotiation model, $\mathbf{N}$, is not known *a priori*, a Negotiated Learning agent also needs to explore the edge parameters, (c, τ, x), in addition to the entity features. In each negotiation, the initiating agent specifies a budget which may be lower than the cost charged by the responding agent, in which case the negotiation fails and the initiating agent may be charged a *failed negotiation fee*. By setting $\gamma < 1$, a Negotiated Learning agent may bid less than its potential benefit, but if the negotiation fails, it can then gradually increase γ.[6]

- **Attraction benefit threshold,** ξ: When two or more Attraction means are interleaved for a period of time, a Negotiated Learning agent is subject to rapidly switching amongst the associated entities, or capacity profiles in this example. So, a value of ξ greater than zero increases the *stickiness* of the entity selection and avoids excessive switching.

  Conversely, this stickiness also implies a delay in exploiting new opportunities. In our example scenario, lower values of ξ yield higher cost savings; however, the appropriate value for ξ in a particular problem depends on the tradeoff between switching costs and the cost of missed opportunities.

Similarly, the impact of the **Attraction update weight** parameters, ω_b and ω_e, also depends on the tradeoff between switching costs and opportunity costs. These parameters control the smoothness of Attraction evolutions. Using the original noise-added capacity aggregate management scenario, Figure 6.18 illustrates the smoother evolutions at lower update weights and more *responsive* evolutions at higher rates. In this specific example, changes in the update weights do not materially impact the cost savings performance. For comparison, in the variable rate selection experiments, higher update weights allow the agent to recognize the fleeting opportunities presented by the irregular variable rate tariffs T4 and T5 of Figure 6.1 more quickly so that it has more time to exploit those opportunities.

[4] Note that this benefit is computed using the Attraction bounds and controls negotiation versus the benefit threshold, ξ, which uses the Attraction means and controls entity selection.

[5] If multiple negotiations are to be invoked in the same time step, γ could be used as a factor in computing the budget for each negotiation, each weighted by its probability of success.

[6] If the failed negotiation fees are high, the agent may benefit from keeping $\gamma = 1$.

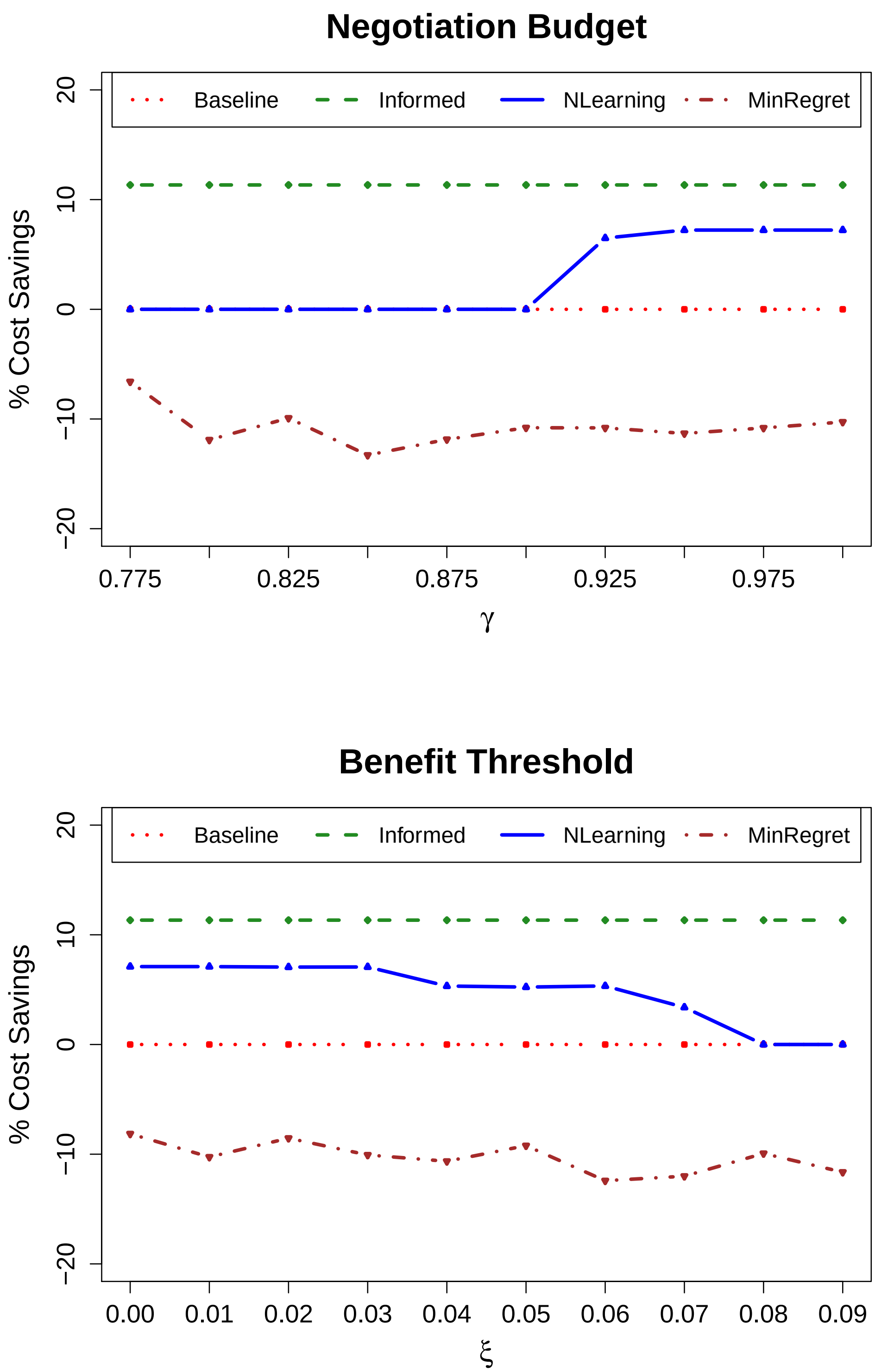


Figure 6.17: Sensitivity of percentage cost savings in response to increasing values of (a) negotiation budget factor γ, and (b) Attraction benefit threshold ξ.

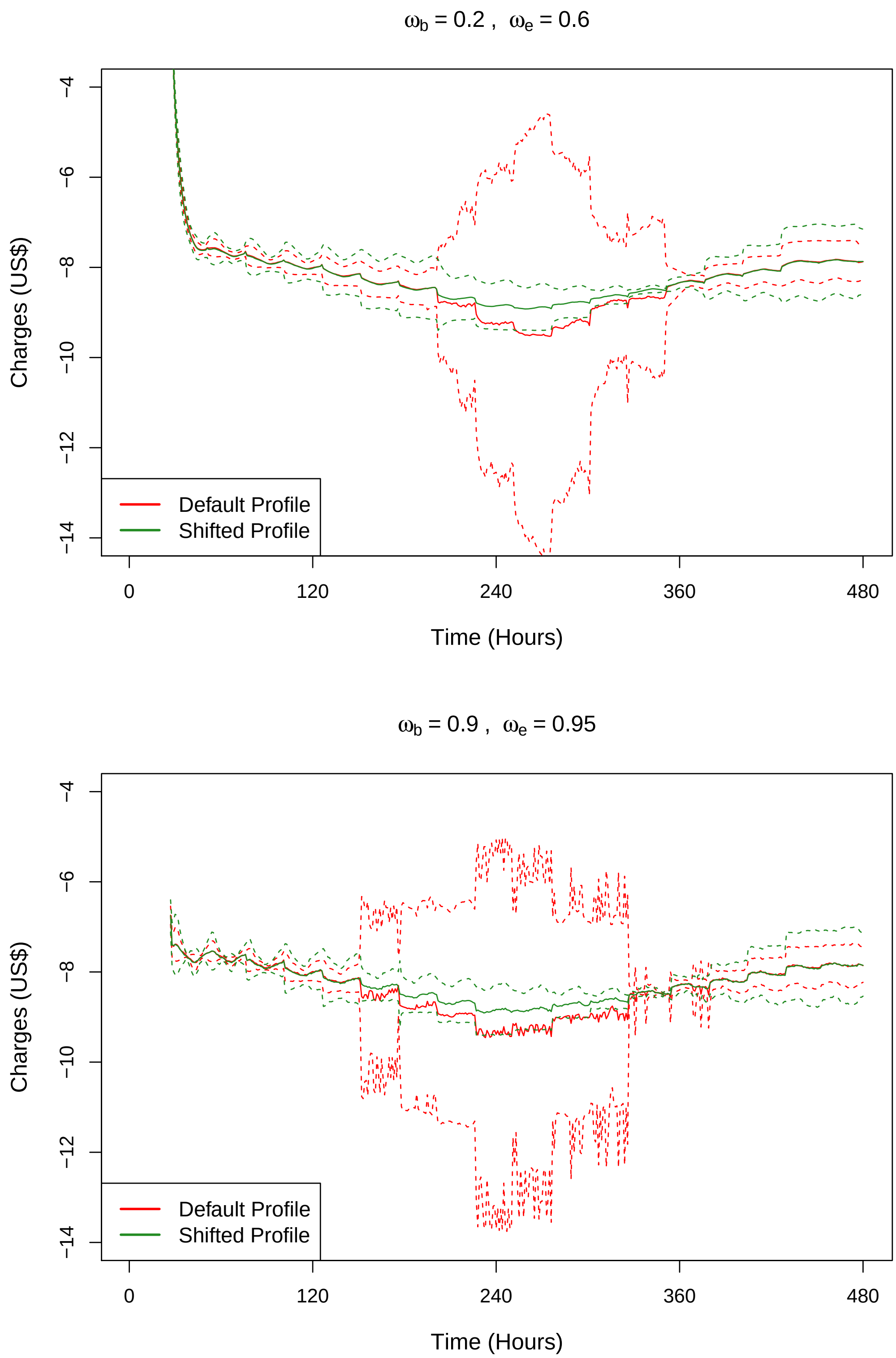


Figure 6.18: Comparison of the smoothness of evolution of Attraction means and bounds with low (top) and high (bottom) update weights ω_b and ω_e.

6.2.2 Scalability Experiments

Under this heading, in addition to our focus on scaling with the number of agents in the environment, we also address the scalability of multiple shifted profiles for an Aggregator agent's controllable capacity and briefly demonstrate model learning and self-play.

- **Number of Component agents**: Figures 5.19-5.21 illustrate the deterioration of information in the Attractions when the negotiation model $\mathbf{N}$ is unknown and the number of component agents increases. The Aggregator agent requires increasing numbers of time steps to recognize the shifting opportunity as can be seen from the delay in the Attractions of two profiles diverging from each other. With 10 agents, the Aggregator completely misses the opportunity because the Attraction means never diverge sufficiently to trigger a profile switch. This is explained by the increasing amounts of time needed for the Aggregator agent to negotiate with the increasing numbers of Volatile agents. If $\mathbf{N}$ is known, the deterioration is observed at larger numbers of agents since the Aggregator agent is able to gather information more efficiently.

  The specific number of agents needed to cause the Aggregator agent to miss opportunities depends on the length of time for which the opportunities present themselves and the fraction of agents from which information needs to be gathered to recognize the opportunities. For example, if an opportunity is short-lived but only 2 of the 16 Volatile agents needs to be interrogated to recognize the opportunity, the Aggregator agent is more likely to succeed in doing so than if all 16 agents need to be interrogated. This observation also leads to our solution to this scaling limitation: *class-based negotiations* instead of agent-based negotiations.

  The agent class model, $\mathbf{K}$, allows us to generalize information from a sample of agents within that class to all the agents in the class. So, if the Aggregator agent applies this generalization when computing the expected aggregates, it is more likely to recognize the opportunities for cost savings. This hypothesis is borne out experimentally as shown in the top subfigure of Figure 6.22. We observe that the basic agent-based negotiation model, formally a 1-to-1 mapping from the set of agents to the set of agent classes, $\mathcal{I} \rightarrow \mathcal{K}$, results in lower savings as the number of agents increases. On the other hand, with the class-based negotiation model, the savings scale consistently.

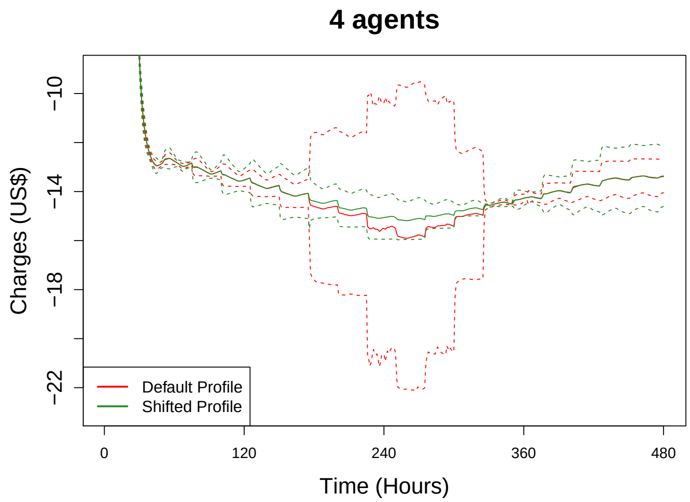


Figure 6.19: With 4 Volatile agents, the Aggregator agent is able to obtain and exploit negotiated information successfully using *agent-based negotiation*.

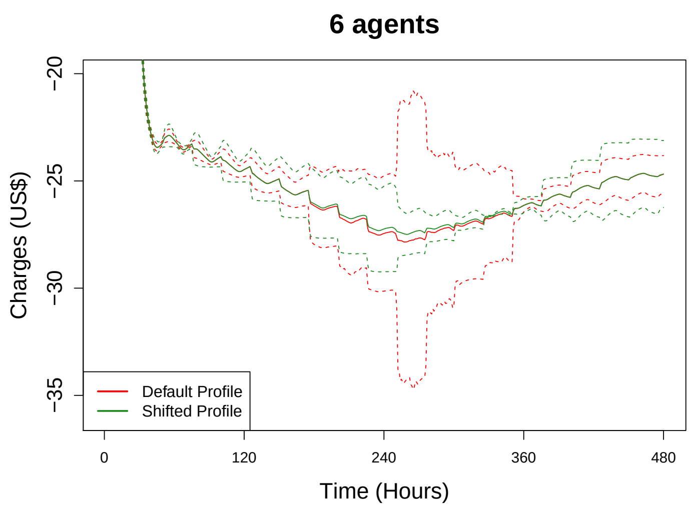


Figure 6.20: With 6 Volatile agents, the Aggregator agent requires more time steps to acquire the negotiated information needed for the Attraction means to sufficiently diverge.

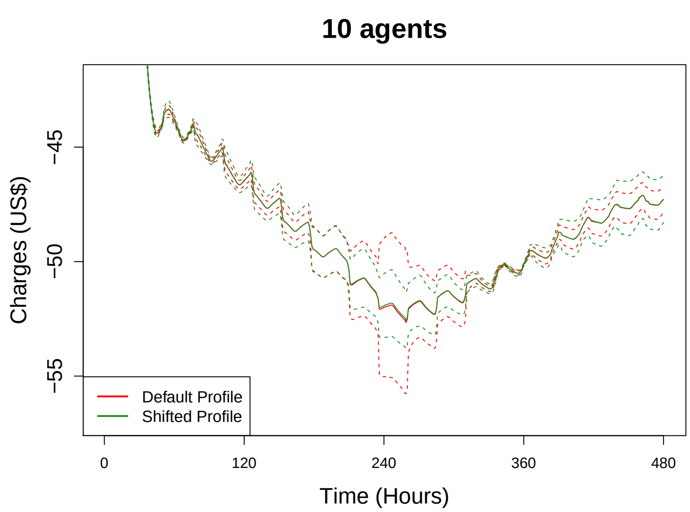


Figure 6.21: With 10 Volatile agents, the Aggregator agent is unable to acquire enough negotiated information using *agent-based negotiations* to capture the shifting opportunity.

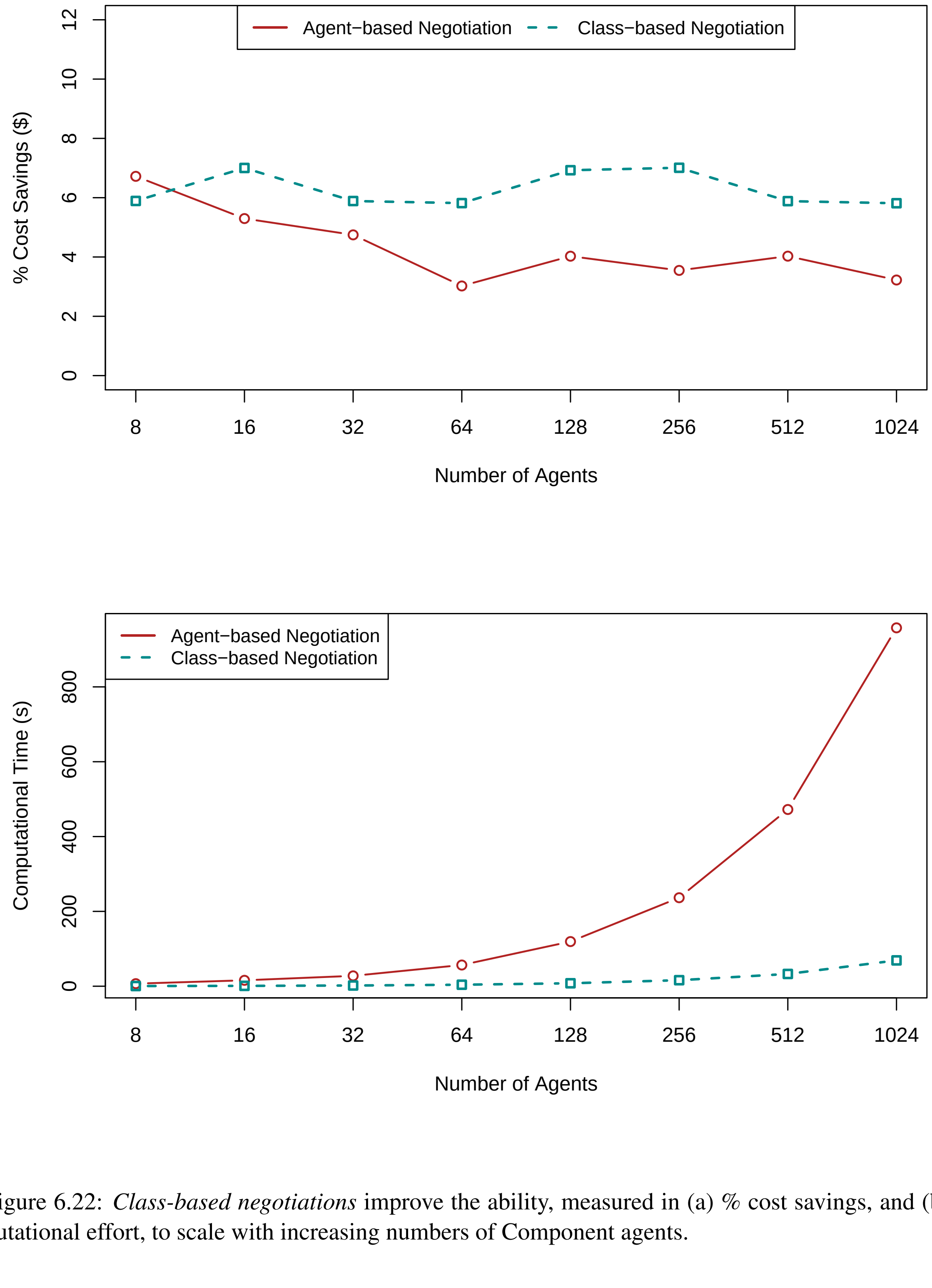


Figure 6.22: *Class-based negotiations* improve the ability, measured in (a) % cost savings, and (b) computational effort, to scale with increasing numbers of Component agents.

The bottom subfigure of Figure 6.22 shows a more dramatic benefit of the class-based negotiation model. The amount of computational time required to make profile selection decisions increases exponentially with the number of agents due to:

(a) The invocations of the imputation and forecasting methods needed to generate forecasts for each agent's capacity.

(b) The combinatorial growth of negotiation actions in the 0-1 program that chooses negotiations to invoke given the current negotiation budget, ψ.

By generalizing agent information over the whole class, both of the above concerns are mitigated, thus leading to significantly higher computational scalability.

- **Number of shifted profiles**: In the capacity aggregate management scenario, the number of shifted profiles, ρ, available to the controllable capacity of the Aggregator agent, D, is another dimension for scalability. Of particular concern is the possibility of a combinatorial problem with the number of agents, $|\mathcal{I}|$. The Aggregator agent is a singleton by definition, so we only need to concern ourselves with the number of Component agents, $|\mathcal{I} \setminus D|$. Within the ATTRACTION-BOUNDED-LEARNING algorithm, Attractions are computed as follows:

  1. Impute and forecast the entity features derived from other agents, *i.e.*, the Component agent capacity histories. The Component forecasts are independent of the controllable capacity profile, so they are computed once, aggregated, and used repeatedly in the next step.
  2. Aggregate the *Component aggregate* with each controllable capacity profile, $\rho \in P$, to construct $|P|$ *capacity aggregate* entities.
  3. Evaluate the $|P|$ capacity aggregates against the expected reward function, *i.e.*, dot product with the tariff's tiered price forecasts and add shifting penalties and switching costs if applicable, and then select the capacity aggregate entity with the best Attraction mean.

  At worst this procedure scales linearly with $|P|$, which should alleviate concern for scalability along this dimension.

- **Learning the negotiation model**: Figure 6.23 shows single-episode results that illustrate learning of the negotiation model, $\mathbf{N}$, in capacity aggregate management experiments with

4 Volatile agents. The poor performance shown in the top subfigure is the result of misclassifying the Volatile agents, which causes the Aggregator agent to completely miss the opportunity for cost savings. The bottom subfigure shows superior performance and also lower negotiation costs early in the episode reflecting a learned agent classification map in $\mathbf{K}$, and learned (c, τ, x) parameters for the component agent negotiation actions.

In our experimental setup, 10 episodes of training makes a substantial difference in the cost savings performance. We expect that more complex negotiation models will require more training episodes or longer periods of online learning. As an alternative to the online model learning designed into the ATTRACTION-BOUNDED-LEARNING algorithm, one could also use existing techniques for learning the weights of bipartite graphs or factored graphs, *e.g.*, belief propagation, to build the negotiation model through offline training. This approach is only practical when existing data on negotiation histories is available. However, if it is feasible, providing the learned model as a known parameter would help further scalability of Negotiated Learning.

- **Self-Play**: In our final experiments, we create scenarios with 4, 8 or 16 Aggregator agents where each of them is also a component agent from the perspective of the other Aggregator agents. The Aggregator agents are still constrained to their default and shifted profiles and do not exhibit the drifting behavior of the Volatile agents in the environment.

  The tariff tier threshold is configured such that if approximately half of the Aggregator agents adopt a shifted profile, then the aggregate remains under the threshold in a noiseless setting. Indeed, we find that a random half subset of the Aggregator agents choose the shifted profile as illustrated in Figure 6.24. The solid blue lines represent the cost savings for each of 4 Aggregator agents, split evenly by their average savings line.

6.3 Chapter Summary

While our Negotiated Learning technique is a general contribution to machine learning and AI, we have focused its development and evaluation within the context of Smart Grid customer agents. Specifically, we applied Negotiated Learning to the problems of (a) variable rate tariff selection, and (b) capacity aggregate management. Using these examples, we demonstrated through simulation experiments: (i) the value of negotiated information, (ii) the importance of a well-informed negotiation model, (iii) learnability of negotiation models, and (iv) robustness to various configuration parameters.

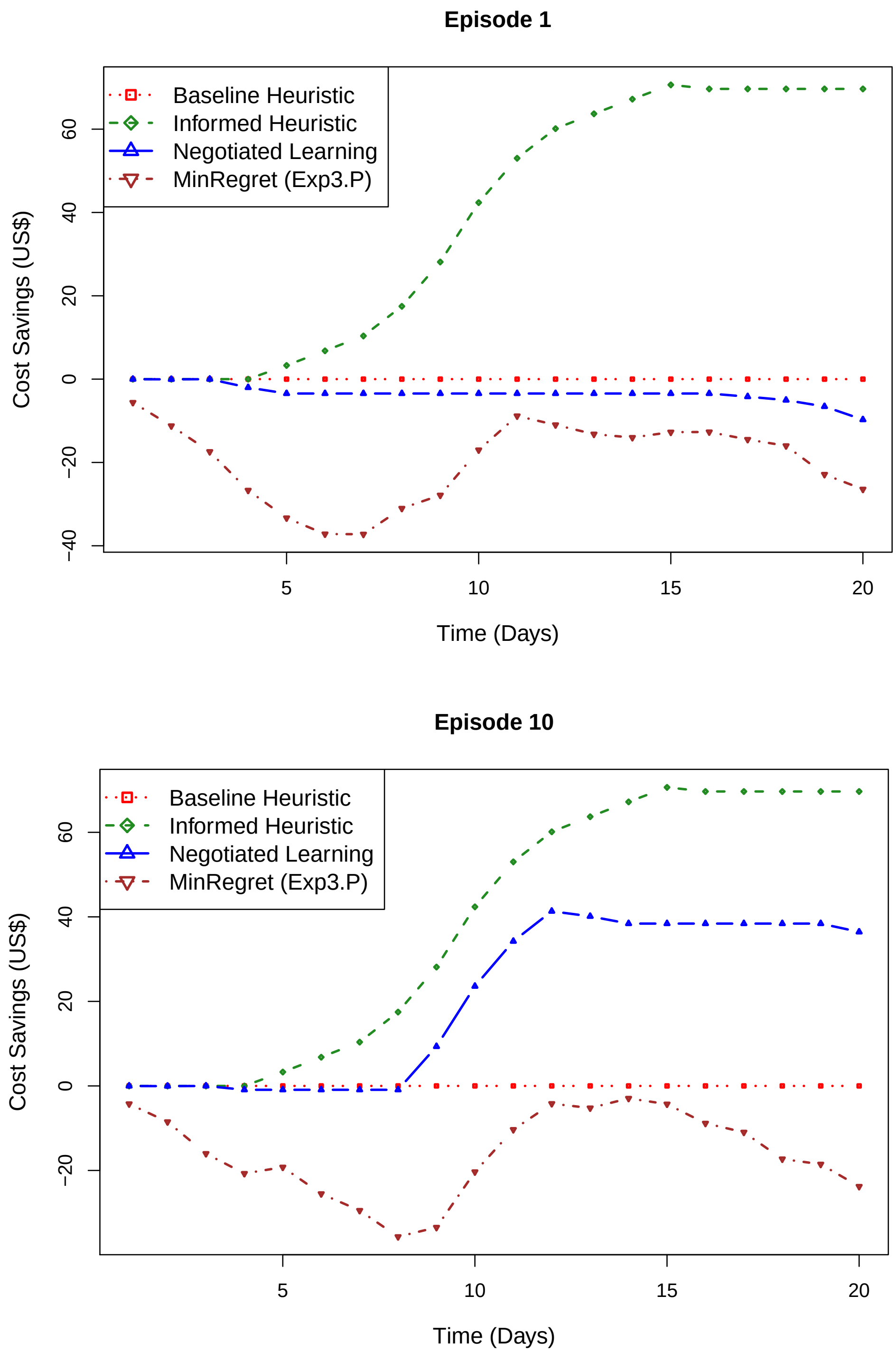


Figure 6.23: Cumulative cost savings in (a) episode 1 when the agent classification map in **K** and the negotiation model **N** are unknown, and (b) episode 10 where **K** and **N** are being learned.

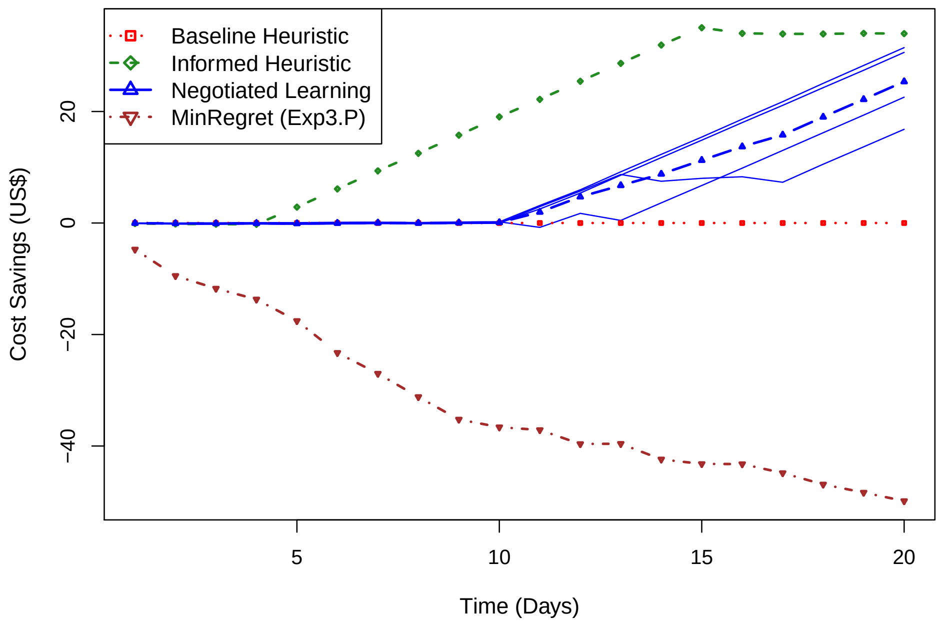


Figure 6.24: Cumulative cost savings in self-play for Aggregator agents using Negotiated Learning.

Many aspects of the Negotiable Entity Selection Process representation and the ATTRACTION-BOUNDED-LEARNING algorithm are inspired by or build upon previous scientific contributions across an array of disciplines. We explore these influences and other related work, and explain our relative positioning, in the next chapter.

Chapter 7

Related Work

The focus of this thesis lies at the intersection of a number of fields including computational energy sustainability, machine learning, multiagent systems, and behavioral economics. In this chapter, we briefly survey relevant research from these fields. While we include applications of machine learning and AI in the Smart Grid domain, we do not survey applications of these disciplines to other domains – a vast undertaking that is beyond the scope of this document.

Section 7.1 provides an overview of research on Smart Grid agents. Section 7.2 identifies relevant aspects of single agent online learning. Section 7.3 delves into multiagent models and algorithms for planning, learning, and strategic decision-making. We include perspectives on how our work relates to prior research within each section, but we also highlight key aspects of our relative positioning in Section 7.4.

7.1 Smart Grid Agents

Russell and Norvig characterize task environments as simpler if they are *fully observable*, *deterministic*, *episodic*, *static*, and *discrete* with a *single agent* [Russell and Norvig, 2003]. Developing Smart Grid customer agents is challenging because the task environments are generally *partially observable*, *stochastic*, *sequential*, *dynamic* and *continuous* with *multiple agents*. Nonetheless, this is a challenge that must be tackled aggressively [Gomes, 2009].

7.1.1 Market Design and Efficiency

Mechanism Design: The benefits of a Smart(er) Grid have been clearly laid out in academic research [Gellings et al., 2004] [Amin and Wollenberg, 2005] as well as field studies [Kannberg

et al., 2003] [NREL, 2012] [Faruqui and Palmer, 2012]. Given the rapid evolution of the regulatory environment, much research has focused on the design of wholesale power markets [Sun and Tesfatsion, 2007] [Contreras et al., 2001] [Liao et al., 2010] so that major disruptions such as the California Energy Crisis [Borenstein, 2002] can be avoided in future. Tariff markets are substantially different in that various segments of customers have diverse preferences and the population therefore tends to distribute demand across many suppliers.

Energy Efficiency: Other research explores opportunities for energy savings through efficiency programs such as demand response (DR) [Hammerstrom, 2008] and demand side management (DSM) [Loughran and Kulick, 2004] [Strbac, 2008] [Blackhurst et al., 2011]. [Hart, 2008], [Kolter et al., 2010] and [Kolter and Ferreira, 2011] are examples of research on monitoring and prediction infrastructure for energy usage in buildings. DR relies on tariff rate structures that incentivize customers to curtail or shift demand when explicitly instructed to do so by the distribution utility. We focus instead on DSM where active customer participation in managing energy usage is encouraged through economic incentives. Rate structures such as TOU, CPP and RTP have been studied to some extent through limited field deployments [Barbose et al., 2005].

7.1.2 Agent Simulation and Strategies

Agent Simulation: Introducing changes in the power grid is very difficult given the criticality of the system and the inability to replicate a live test environment. Therefore, significant research effort has gone into power grid simulation technology. While power systems research has largely taken a control-theoretic approach, *e.g.*, [Allen et al., 2001], agent-based simulation [Tesfatsion, 2006] has emerged over the last decade as a viable alternative or extension to previous models [Ketter et al., 2010].

Our work on hierarchical Bayesian time series simulation is based on Seasonal ARIMA models [Cryer and Chan, 2008] and multilevel Bayesian models [Gelman and Hill, 2007]. Examples of autoregressive Bayesian prediction with latent variables are described in [West and Harrison, 1997]. Hierarchical Bayesian models and Dynamic Bayesian Networks (DBNs) have been studied extensively, *e.g.*, [Murphy, 2002]. We apply Gibbs-sampling based inference techniques [Geman and Geman, 1984] as is typical with complex DBNs [Koller and Friedman, 2009].

Models for generating synthetic load profiles for agents representing Smart Grid customers are studied in [Armstrong et al., 2009] and [Reddy and Veloso, 2012] whereas agent simulation models based on real data samples are studied in [Paatero and Lund, 2006], [Hirsch et al., 2010], [Gottwalt et al., 2011], and [Chrysopoulos and Symeonidis, 2009]. Simulation environments

containing all of the various types of agents required to model a distribution grid are described in [Karnouskos and de Holanda, 2009] and [Ketter et al., 2013].

Agent Strategies: Earlier work on agents in the Smart Grid has focused on competitive bidding agents in wholesale power markets from a supplier's perspective; *i.e.*, these agents typically represent large power plants who sell power directly into the wholesale market [David and Wen, 2000]. [Rahimi-Kian et al., 2005] and [Xiong et al., 2002] are examples of such agent strategies that use Q-LEARNING [Watkins and Dayan, 1992]; [David and Wen, 2000] provides a literature survey of similar agents. [Vytelingum et al., 2010] studies the design of trading agent strategies in continuous double auction markets and emergent equilibria from a game-theoretic perspective.

[Braun and Strauss, 2008] describes *commercial aggregators* as contracting trading entities in a sense most similar to our definition of Broker Agents. However, they do not address the possibility of autonomous agents playing that role. To the best of our knowledge, developing reinforcement learning-based strategies for autonomous Broker Agents in Smart Grid Tariff Markets is a novel research agenda.

Research on agents representing Smart Grid customers has emerged only recently. Initial work was directed towards agent strategies that can manage storage capacities like those offered by plugin electric vehicles (PEV). These types of agents need to decide when they should buy power from the grid and when, if at all, to sell it back, in response to dynamic prices offered by *suppliers* (the distribution utility or brokers) [Vytelingum et al., 2011]. [Voice et al., 2011] studies the design of incentives from a supplier's perspective in the presence such storage agents.

Other efforts have addressed coordination of customers to form micro-grids [Braun and Strauss, 2008] and virtual power plants [Chalkiadakis et al., 2011]. In work closely related to ours, [Ramchurn et al., 2011] studies agent-based adaptive control for decentralized demand-side management using the Widrow-Hoff rule [Widrow and Hoff, 1960] under RTP tariffs. Our utility optimizing agent is related to distributed control with constrained reasoning, *e.g.*, [Modi et al., 2005], and also team coaching in adversarial settings, *e.g.*, [Riley, 2005]. Our stochastic approximation algorithm is based on classic multiattribute utility theory [Wellman, 1985] and decision theory [Horvitz et al., 1988]. A broad overview of the AI challenges in Smart Grid agent design is presented in [Ramchurn et al., 2012].

7.2 Agent-based Online Learning

The decision-making responsibilities of an agent in a multiagent environment can always be viewed as a single agent decision-making problem by treating all other agents simply as features

of the *environment*. However, such an approach ignores the potential for immense improvements in decision-making through exploitation of the multiagent structure. On the other hand, capturing the structure in the problem representation leads to more complex models and often more complex algorithms. Navigating the resulting spectrum of solution techniques is the challenge we face in developing learning Smart Grid agents.

7.2.1 Planning and Learning

Single agent techniques for planning and learning have been studied extensively [Russell and Norvig, 2003]. The domain of problems that we encounter in the development of Smart Grid agents is generally complex and poorly defined. Consequently, approaches that rely on well-defined models of the environment are difficult to apply—*e.g.*, state- or model-space planners, model-based reinforcement learning methods like R-MAX [Brafman and Tennenholtz, 2002] and MBIE [Strehl and Littman, 2008].

Temporal difference or TD(λ) algorithms like Q-LEARNING and SARSA [Rummery and Niranjan, 1994] are prominent techniques for reinforcement learning in complex single-agent environments where transition models are unknown or difficult to describe [Sutton and Barto, 1995] [Mitchell, 1997]. We have taken this approach in developing autonomous broker agent strategies.

7.2.2 Regret Minimization

An alternate game-theoretic approach to single agent online learning is based on Hannan consistency [Hannan, 1957] or *no-regret* [Foster and Vohra, 1999], where the learned policy is expected to outperform any fixed strategy. No-regret learning can occur in the *full information model*, where the rewards attributed to all available actions are observable (*e.g.*, experts problem [Littlestone and Warmuth, 1994]), or the *partial information model*, where only the rewards for the executed action are available (*e.g.*, multiarmed bandit problem [Auer et al., 1995]).

Various algorithms such as randomized weighted majority [Littlestone and Warmuth, 1994], regret-matching [Hart and Mas-Colell, 2000], and smooth fictitious play [Fudenberg and Levine, 1995], have been shown to exhibit no-regret [Blum and Mansour, 2007] under the full information assumption. However, since we target semi-cooperative problems, we generally are unable to make the full information assumption.

The UCB2 algorithm is widely used in stochastic multiarmed bandit problems, but we have also focused on problems with dynamic non-stochastic features. So, we have instead used the EXP3 algorithms [Auer et al., 1995] [Auer et al., 2002], which make no stochastic assumptions, for comparison in evaluations of Negotiated Learning.

7.2.3 Active Agent Learning

Negotiated Learning also draws upon work related to *value of information* and *active learning*. Recent work in machine learning has focused on active learning for collecting labeled examples in semi-supervised classification problems (*e.g.*, [Jones et al., 2003]). We tackle a different scenario where we seek information for the purpose of exploration.

This is more similar to information gathering actions in partially observable Markov decision processes (POMDP). For example, in *active perception* problems, a robotic sensor is tasked with the goal of acquiring a particular percept [Bajcsy, 1988]. The planning problem for the robot is to find the lowest cost sequence of actions that would enable the sensor to perceive the necessary information. This evaluation is based on a well-informed model of the environment and the actions are invoked by a planner controlling the robot.

In our Smart Grid agent setting, specifically in our Negotiated Learning approach, we similarly tackle the problem of seeking information with incurred costs. However, our goal is fundamentally different. We only seek to acquire the percept if the *expected information value* of that percept would influence our entity selection decisions. Moreover, we take an online learning approach based on exploration-exploitation instead of planned exploration. In active perception, since perception itself is the *goal*, the robot does not have to evaluate what else it could be doing that would maximize its *reward*. So, unlike in both EXP3 and ATTRACTION-BOUNDED-LEARNING, there is no evaluation in active perception of the opportunity costs incurred by pursuing exploration instead of exploitation.

7.3 Multiagent Models and Algorithms

The problem of action selection in multiagent systems can be viewed from many perspectives:

1. goal achievement through model-based multiagent coordination (planning),
2. reward maximization based on beliefs and experience (online/reinforcement learning), and
3. reward maximization in response to the actions of other agents (game theory).

7.3.1 Planning with Partial Observability

Partial observability in cooperative model-based agent domains has been studied primarily through the framework of POMDPs, *e.g.*, [Kaelbling et al., 1998]. Dec-POMDPs extend POMDPs to

multiagent settings but are computationally prohibitive in most cases [Borenstein, 2002]. Therefore, much research has focused on exploiting multiagent structure to simplify the problem. For example, [Roth et al., 2007] and [Melo and Veloso, 2009] limit communication and agent modeling to subsets of states where interaction is important, whereas [Witwicki and Durfee, 2010] and [Oliehoek et al., 2012] quantify *influences* between agents to constrain the problem. While Dec-POMDPs are aimed at deriving joint policies for all agents, I-POMDPs [Gmytrasiewicz and Doshi, 2005] model the multiagent environment from a single agent's perspective in order to derive a policy for only that agent. A key aspect of our Negotiable Entity Selection Process representation is that it trades off some generality to expose elements of multiagent structure that are lost in other representations.

Asking for Information: [Armstrong-Crews and Veloso, 2007] introduced Oracular POMDPs (O-POMDPs) as another mechanism to address the computational complexity of POMDPs. O-POMDPs assume that there is an *oracle* that can reveal the full state information when requested [Armstrong-Crews and Veloso, 2008]. This is a key principle in our Negotiated Learning approach whereby we assume that all other agents together form an oracle.

Semi-Cooperative Planning: Complex multiagent domains like the Smart Grid give rise to semi-cooperative problems because of the combinations of self-interests and joint interests for the agents involved. [Powell et al., 2011] presents an approach to semi-cooperative planning where they decompose a sequential resource allocation problem into a series of subproblems such that different agents are responsible for coordinating different subproblems over time. They demonstrate in simulation experiments that such an approach to distributed semi-cooperative control approaches the performance of centralized control through a single agent. However, the approach is based on solving a known system dynamics model using approximate dynamic programming, so it is difficult to apply in the Smart Grid. We instead take a probabilistic approach to behavioral modeling in our stochastic utility optimizer for adaptive capacity management.

7.3.2 Multiagent Reinforcement Learning

[Stone and Veloso, 2000] survey multiagent systems from a machine learning perspective. In multiagent reinforcement learning, as with POMDPs, exploiting the hierarchical or factored structure of the problem to build rich models yields improvements in the complexity of addressed problems [Dietterich, 1999] [Guestrin et al., 2003], or in the rate of convergence [Bradtke and

Barto, 1996][Boyan, 1999]. In problems with high-dimensional states, feature selection techniques using regularized regression or predictive state representations [Singh et al., 2004] have also been fruitful, *e.g.*, LARS-TD [Kolter and Ng, 2009] and PSTD [Boots and Gordon, 2011].

7.3.2.1 Experience-Weighted Attraction Learning

In behavioral game theory, [Camerer and Ho, 1999] introduced Experience-Weighted Attraction learning (EWA) as a solution technique for sequential decision-making in repeated agent interactions. EWA presents a hybrid model that generalizes reinforcement learning and *belief learning*. Belief learning dates back to Cournot learning [Cournot, 1838], which prescribes that an agent choose the best response to the actions chosen by the other agents at the previous time step.

In contrast, *fictitious play* [Brown, 1951] suggests that the agent best respond to the mixed-strategies of each agent as observed from all of their previous action choices. Weighted fictitious play [Cheung and Friedman, 1997] generalizes from these two extremes by assigning more weight to the recent actions of the other players and best responding to the resulting weighted mixed-strategies. Thus, weighted fictitious play is parameterized on ϕ in $\mathbb{R}[0, 1]$ such that we get Cournot learning when ϕ=0 and the original unweighted fictitious play when ϕ=0. Belief learning is related to no-regret learning, although it focuses on best-response dynamics instead of Hannan consistency or regret minimization in general.

In reinforcement learning, an agent only uses information about its own history of actions and received rewards to compute state-action values.[1] In belief learning, the agent ignores the history of its actions and instead computes beliefs about *forgone rewards* associated with the actions it *did not* execute and uses those beliefs as updates to state-action values. However, experimental studies have shown that received rewards and beliefs about forgone rewards are both used simultaneously by human subjects when choosing actions [Erev and Roth, 1998].

Therefore, EWA uses both types of information to compute *attractions* for each action choice. Updates to attraction values are parameterized on δ and κ both in $\mathbb{R}[0, 1]$ such that when δ=0 and κ=1, EWA is equivalent to reinforcement learning with cumulative rewards; for δ=0 and κ=0, to reinforcement learning with average rewards; and for δ=1 and κ=0, to weighted fictitious play. These relationships are illustrated in Figure 7.1. [Camerer, 2008] present experiments where they show that EWA with intermediate values for δ, *i.e.*, a true hybrid configuration, outperforms both reinforcement and belief learning. Those experiments also show that different subjects weigh received and forgone payoffs differently thus requiring different δ values.

[1]We use the terminology of *actions* and *rewards* instead of the game-theoretic terms *strategies* and *payoffs*.

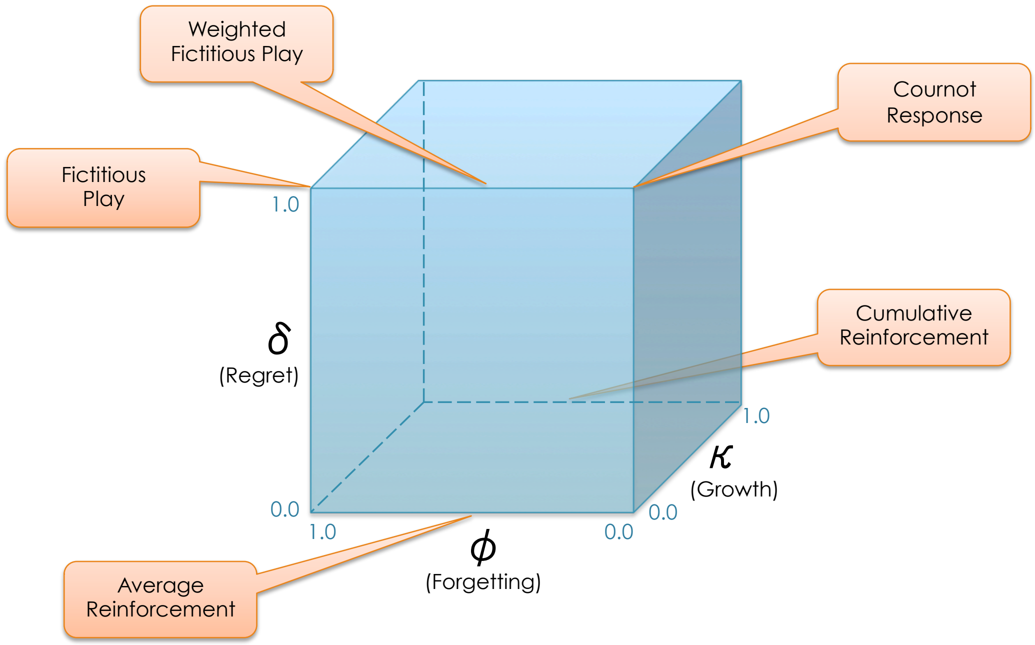


Figure 7.1: EWA learning is a generalization of well-known online learning methods that can be obtained by using specific values for its δ, κ, and ϕ parameters.

Table 7.1 shows the information requirements of various learning theories. Imitation learning, where agents imitate the best-performing agents is also related to no-regret learning [Ross et al., 2011], but it requires information about other agents' rewards. Anticipatory learning which tries to model the learning behavior of other agents has even higher information requirements [Chong et al., 2006]. This table highlights the opportunities for a Smart Grid agent to use distinct learning theories by negotiating for the information that it does not intrinsically possess.

7.3.3 Strategic Decision-Making

Continuing on to research in game theoretic approaches to multiagent systems that are unrelated to reinforcement learning, we present a few additional influences on our work.

7.3.3.1 Multiscale Decision-Making

The need for multiscale analysis has become increasingly important as information at macro and micro scales needs to be considered simultaneously to solve optimization problems in various domains including climate studies, information networks and power systems [Dolbow et al., 2004].

Table 7.1: Minimal information used by various learning theories. *[Camerer, 2003]*

Information	Reinforcement	Belief	EWA	Imitation	Anticipatory
i's action choice	✓		✓		
$-i$'s action choices		✓	✓	✓	✓
i's received reward	✓	✓	✓		
i's forgone reward		✓	✓		✓
$-i$'s received rewards				✓	✓
$-i$'s forgone rewards					✓

Sutton, *et al.*, introduced Semi-Markov Decision Processes (SMDP) as a formalism to incorporate multiple levels of temporal abstraction into a reinforcement learning framework through the use of subgoals and *options* that can be integrated with primitive MDP actions.

Wernz and Deshmukh [Wernz and Deshmukh, 2010b] introduce *multiscale decision-making* by extending the problem to multiple dimensions. They study organizational decision-making with temporal and hierarchical dimensions using game-theoretic tools. We adopt a similar approach in [Reddy and Veloso, 2013] to formulate the Smart Grid customer agent problem with temporal and contextual dimensions although we propose to use different solution techniques.

7.3.3.2 Graphical Representation

Influence diagrams [Howard and Matheson, 1984] introduce a decision-theoretic approach to agent design by modeling decision problems as Bayesian networks containing chance, decision and utility nodes. MAIDs [Koller and Milch, 2003] extend this approach to multiagent problems.

Graphical games [Kearns et al., 1995] and action-graph games [Shoham and Leyton-Brown, 2009] combine influence diagrams with game-theoretic models to constrain strategy spaces that need to be evaluated. While MAIDs adopt a descriptive approach towards analyzing stochastic multiagent systems, Interactive Dynamic Influence Diagrams (I-DIDs) [Doshi et al., 2008] adopt a prescriptive approach by modeling the multiagent system from a single agent's perspective; *i.e.*, I-DIDs are to I-POMDPs as MAIDs are to POMDPs.

7.3.3.3 Stochastic Games

While Dec-POMDPs primarily address cooperative problems, game theoretic models are often used to tackle adversarial multiagent problems [Shoham and Leyton-Brown, 2009]. Stochastic games (SG) [Shapley, 1953], *a.k.a.* Markov games, are the primary abstraction for this approach. In such games, the joint actions of all agents result in stochastically determined stage games at each time step. An MDP can be viewed as a single agent SG. Thus, model-free MDP-based algo-

rithms like Q-LEARNING have been generalized to SG-equivalents, *e.g.*, MINIMAX-Q [Littman, 1994], NASH-Q [Hu and Wellman, 1998] [Hu and Wellman, 2003], and CE-Q [Greenwald and Hall, 2003] [Murray and Gordon, 2007].

Algorithms such as WOLF-IGA [Bowling and Veloso, 2003] and AWESOME [Conitzer and Sandholm, 2006] are designed to adapt to other learning agents in dynamic environments. Partially observable stochastic games (POSG) offer a rich formalism that generalizes both stochastic games and POMDPs, but they are extremely difficult to solve in general [Hansen et al., 2004].

7.3.3.4 Semi-Cooperative Negotiation

Preferences and incentives can be interpreted as internal versus external motivation for an agent. In coalition formation [Sandholm and Lesser, 1995] [Sandholm and Crites, 1996], agents are typically intrinsically motivated, *e.g.*, [Klusch and Gerber, 2002][Konishi and Ray, 2003]. Preference-handling, especially in the context of coalition formation, has been studied extensively, *e.g.*, [Wellman, 1985] [Brafman and Domshlak, 2008]. [Boutilier, 1996] studies the design of mechanisms for planning, learning and coordination in semi-cooperative multiagent systems based on conventions and social laws.

Literature on incentives is mostly focused on issues of labor compensation, *e.g.*, [Prendergast, 1999], or mechanism design, *e.g.*, [Groves, 1973]. Incentives can be categorized as *financial*, *moral*, or *coercive*. While financial incentives easily translate to costs, we can also view the effort involved in *encouraging* or *forcing* other agents to deviate from their natural preferences as having costs too. We can reverse the view of the cost of *offering* incentives to see it as the cost of *receiving* information, thus allowing us to draw on the principles of information value theory [Howard, 1966]. Specifically, a negotiating agent would want to incur the costs of offering incentives only if their acceptance by other agents causes a notable change in expected reward from the environment.

An agent in incentive-based negotiation must decide when and what incentive to offer to other agents. Similarly, research in learning from demonstration studies when and what information to request, *e.g.*, [Riley, 2005][Chernova and Veloso, 2009]. Reddi and Brunskill [Reddi and Brunskill, 2012] recently introduced Incentive Decision Processes to study problems where an agent offers incentives to reduce costs due to the decisions of another greedy agent. A novel aspect that we tackle in addition is the decision on the choice of agent(s) to whom incentives should be offered.

Crawford and Veloso [Crawford and Veloso, 2008] study negotiation in semi-cooperative agreement problems. They note that agents necessarily reveal information about their own pref-

erences and constraints as they negotiate agreements and show how agents can use this limited and noisy information to learn to negotiate more effectively.

7.3.3.5 Quantal Cognitive Hierarchies

Behavioral game-theoretic models focus on sources of uncertainty due to cognitive biases and constrained computational capabilities amongst agents. Two key concepts emerge: (i) *cost-proportional errors*, and (ii) *iterated strategic reasoning*. Agents make cost-proportional errors if their rate of making errors increases as errors become less costly. This can be modeled by assuming that agents best respond quantally, *e.g.*, using a *logit choice* model, rather than via strict maximization of action values.[2] When each agent responds quantally, the emergent *quantal response equilibrium* (QRE) [McKelvey and Palfrey, 1995] is the generalization of a Nash equilibrium (*i.e.*, Nash equilibria are QRE where all agents act rationally).

In level-k iterated reasoning, an agent assumes that other agents reason at level-j where j=k-1 and level-0 agents act randomly. Quantal Level-k models [Stahl and Wilson, 1995] combine quantal response with level-k reasoning. Camerer [Camerer, 2003] introduced Cognitive Hierarchy (CH) models which assume that a level-k agent encounters a distribution of level-i agents where i ranges from 0 to k-1 and best responds accordingly. Wright and Leyton-Brown combine quantal response and cognitive hierarchical reasoning in QCH models and demonstrate their better fit on experimental data [Wright and Leyton-Brown, 2010]. Such reasoning is the basis of the ϵ-QRE approach in our stochastic utility optimizer for adaptive capacity management.

7.4 How Our Work Fits

Figure 7.2 illustrates the relative positioning of a few of the many representation models that we have surveyed, using three dimensions:

- Fully observable *vs.* partially observable
- Single agent *vs.* multiple agents
- Cooperative *vs.* adversarial

Our focus positions our contributions, especially the use of Negotiable Entity Selection Processes for Negotiated Learning, approximately at the center of the cube.

[2] Multinomial logit choice is equivalent to a Boltzmann distribution model with the *temperature* parameter in the latter model serving as an inversion of the *rationality* parameter in the logit choice model.

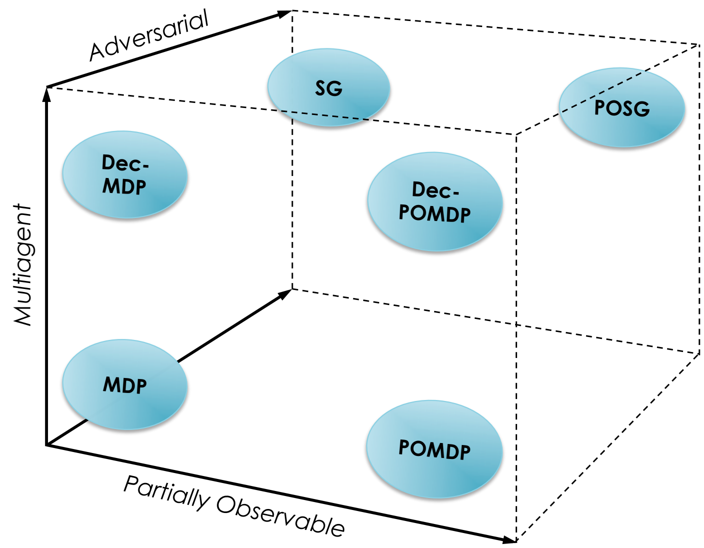


Figure 7.2: Well-known problem representation models positioned relative to the dimensions of partial observability, multiagent interactions, and adversarial interactions.

[Crawford and Veloso, 2008] demonstrate the elicitation of hidden attributes about neighbors through semi-cooperative negotiations, but they do not consider cost/payments or quality of information. Our use of *Attractions* in Negotiated Learning is based largely on [Camerer and Ho, 1999] who propose a behavioral framework that combines reinforcement learning and belief learning to learn from own experiences as well as beliefs about other agents' experiences. Cognitive hierarchies as a partition of agent populations with different levels of reasoning capability, which rationalizes our use of agent classes, is also due to [Camerer, 2008]. Our use of upper and lower bounds in Attractions draws upon model-based interval estimation [Strehl and Littman, 2008] and the principle of communicating only when acquired information may change the agent's policy action [Roth, 2003]. However, none of these works combine negotiating for paid information and simultaneously learning a negotiation model to address partial observability.

Chapter 8

Conclusion

The inexorable growth of human society's reliance on electricity to power our modern lifestyles presents opportunities and challenges of imposing scale. From mobile phones to electric cars, we are steadily introducing new and revolutionary ways to work and play. The opportunity to reduce our dependence on fossil fuels is exciting, but the inherent challenges in enhancing our power generation and distribution infrastructure to create a "Smart Grid" are daunting.

Against that backdrop, this thesis presents original and substantial contributions to research:

(i) in the emerging discipline of *computational energy sustainability* where we address challenges in the design and development of Smart Grid agents, and

(ii) in machine learning and artificial intelligence where we introduce models and learning algorithms for autonomous agents in semi-cooperative environments like the Smart Grid.

In this concluding chapter, we summarize the contributions of this thesis and identify some future directions for research.

8.1 Thesis Contributions

In this thesis we set out to address the following question:

> *How can the multiscale decision-making tasks of a Smart Grid customer be addressed by an autonomous learning agent in a distributed agent environment?*

The goal of this thesis is to demonstrate that the complex decision-making tasks of Smart Grid participants can be addressed effectively by **semi-cooperative learning agents**. Our approach towards semi-cooperative learning is exemplified by our Negotiated Learning technique, where

we first categorize an agent's interactions with other agents in the environment as adversarial or semi-cooperative, and then develop learning strategies that enable the agent to use negotiated information obtained from semi-cooperative agents to devise better strategies against the adversarial agents. While we focus our development and validation of this thesis on Smart Grid agents, we expect that the contributions towards semi-cooperative learning are valuable in other domains that consist of heterogeneous self-interested agents in complex dynamic environments.

We develop models and algorithms for agents in Smart Grid tariff markets where competitive *brokers* sell power to or buy power from *customers* who primarily consume electricity but may also be capable of producing electricity from distributed and renewable energy sources. Customers must select amongst numerous tariffs offered by the brokers and simultaneously manage how much electricity they consume or produce at what time. We categorize these customer decision-making tasks as *tariff selection* and *capacity management*. We address multiple variants of these tasks for specific agent scenarios and also address related problems in the Smart Grid domain such as broker tariff pricing and customer model simulation.

In the remainder of this section, we highlight these problem formulations, along with corresponding representation models and algorithmic methods. We also highlight our analysis of real data from the current power grid, which provides a basis for the simulation experiments that validate our algorithms. These contributions are also summarized in Table 8.1.

1. **Problem formulations**:

   In **Chapter 2**, we explored the problem of developing tariff pricing strategies for broker agents in Smart Grid tariff markets. Brokers are subject to balancing penalties imposed on supply-demand imbalances in their customer portfolio. We formalized the tariff market domain representation and the profit-maximizing goal of a broker agent. We formulated a scalable MDP for the broker agent's decision task, which includes a set of independently applicable pricing tactics and novel domain-specific state aggregation heuristics.

   In **Chapter 3**, we highlighted the importance of software-based simulation in the evaluation of new market structures and agent behaviors for future Smart Grid markets. Power TAC offers a distributed agent-based simulation environment to facilitate such evaluation. We encountered the problem of simulating a long range time series using a combination of offline and online data. This problem is challenging because there is no access to ground truth data that can be revealed over time. This constraint implies that the forecasts must themselves be used as historical values, which leads to a multiplicative effect on any forecasting errors that worsens with the length of the simulation. We use hierarchical Bayesian models to estimate factors needed to mitigate the impact of the multiplicative errors and

include in our problem formulation a requirement for the ability to inject prior beliefs over those factors. While we formulated this problem within the context of Smart Grid customer agent simulation, similar situations arise in other simulation domains.

In **Chapter 4**, we structured the tariff selection and capacity management decision-making responsibilities of Smart Grid customers using a multiscale decision-making framework. We identified challenges in simulating a large variety of heterogeneous Smart Grid customers at various levels of agent granularity. We formulated the *adaptive capacity management* problem faced by customers in multi-dwelling environments like rural electric cooperatives, which impose inherent tradeoffs in self-interest and joint interests amongst groups of semi-cooperative customers who want to achieve individual cost savings as well as shared goals such as reduced variance of their combined capacity levels and intelligent adaptive *capacity shifting* in response to dynamic tariff prices.

In **Chapter 5**, we identified a general class of *negotiable entity selection* problems that are suitable for our Negotiated Learning approach. This class of problems is exemplified by the customer's *variable rate tariff selection* problem where the dynamic prices of the competitive tariffs are published only to currently subscribed customers thus making it challenging for other potential customers to evaluate those tariffs. We also formulated the *capacity aggregate management* problem where a customer is responsible for choosing a profile for controllable capacity based on the expected capacity levels of several component capacities such that the aggregate of the controllable and component capacities does not cross a threshold that triggers less favorable tariff prices. In Section 5.4, we also identified a few problems beyond the Smart Grid domain that are suitable for Negotiated Learning and provided a detailed formulation for one of them—the *TV viewer* problem.

2. **Representation models**:

   In **Chapter 4**, we addressed the need for a versatile customer representation model for large scale simulation with our *factored customer model* framework. The framework introduces capacity originators, tariff subscribers and utility optimizers as decision-theoretic components of a Smart Grid customer and defines their interactions using a multiscale decision-making framework. The behavior of each component is determined by a set of configurable factors that are sufficiently general so that a wide variety of customers can be instantiated using appropriate settings for the factors. We demonstrate the versatility of the framework by citing numerous factored customer model instances created for Power TAC tournaments. Moreover, factored customer models enable a utility optimizer to represent the semi-cooperative interactions of Smart Grid customer agents in the adaptive capacity

Table 8.1: Summary of thesis contributions.

Category	Contribution	Chapter
Problem formulations	– Broker agent tariff pricing Markov decision process	2
	– Scalable customer model time series simulation	3
	– Multiscale Smart Grid customer decision-making	4
	– Customer agent adaptive capacity management	4
	– Customer agent variable rate tariff selection	5
	– Customer agent capacity aggregate management	5
Representation models	– Factored customer model framework and instances	4
	– Negotiable Entity Selection Processes (NESP)	5
Algorithmic methods	– Bayesian long range time series forecasting	3
	– Semi-cooperative stochastic utility optimization	4
	– Negotiated Learning	5
	– ATTRACTION-BOUNDED-LEARNING	5
Empirical evaluation	– Ontario IESO wholesale prices analysis	2
	– Experiments to validate our algorithmic methods	*

management problem, and for tariff subscribers and capacity originators to do the same for the variable rate tariff selection and capacity aggregate management problems respectively.

In **Chapter 5**, we introduced *Negotiable Entity Selection Processes* (NESP), which expose the semi-cooperative structure of agents in the variable rate tariff selection and capacity aggregate management problems. While these problems exemplify the use of NESP representations, the NESP model itself is applicable in other semi-cooperative agent domains as we described in Section 5.4. The NESP is a novel representation that captures *negotiable partial observability*—the semi-cooperative multiagent structure that is exploited by Negotiated Learning. Key elements of the structure are represented by the agent classification model $\mathbf{K}$ that maps the set of agents $\mathcal{I}$ to a set of classes $\mathcal{K}$, the state transform function $\varphi(s(t), \mathcal{F})$ that generates reduced-uncertainty states for a given state $s(t)$ along the feature dimensions $\mathcal{F}$, and the negotiation model $\mathbf{N}$ which creates a bipartite graph mapping transformed states to agent classes where each edge of the graph represents a possible negotiation action. In a fundamental deviation from Markov decision processes, the NESP allows an agent to select one entity selection action and zero or more negotiation actions at each time step, thus facilitating *simultaneous* exploration and exploitation.

3. **Algorithmic methods**:

In **Chapter 3**, we addressed the long range customer model time series simulation problem using our *augmented hierarchical Bayesian methodology*. We used data from a fine-grained

household consumption model to learn how a coarse-grained model can simulate a time series that approximately replicates the essential characteristics of the given data. Instead of trying to recover the true parameters that were used to generate the fine-grained data: (i) we modeled more general factors that represent a broad set of customers, (ii) determined appropriate hierarchical models based on those factors, (iii) estimated the coefficients or distributions for those factors, and (iv) used the factor estimates to add an augmentation term to the simulation forecast that minimizes the multiplicative forecasting errors.

In **Chapter 4**, we developed our *stochastic utility optimization* algorithm, which enables the computation of approximate quantal response equilibrium, ϵ-QRE. This approach models the reactivity, receptivity and rationality of individual customers to measure their responsiveness to recommendations for capacity shifting. The recommendations are computed using a logit quantal capacity management model based on a self utility function for each customer that considers cost savings from capacity shifting, the disutility of shifting, and the joint utility with neighboring agents of a shifted capacity profile.

In **Chapter 5**, we introduced our *Negotiated Learning* technique, and the ATTRACTION-BOUNDED-LEARNING algorithm in particular, as an effective mechanism to address the exploration and exploitation needs of decision-making agents in dynamic semi-cooperative environments. A key factor in the success of our approach is in recognizing the importance of separating, where feasible, the decision-making criteria for exploration and exploitation. As a critical element in the design of ATTRACTION-BOUNDED-LEARNING, we use *Attractions* to separately capture metrics for negotiation (exploration) and entity selection (exploitation). Moreover, we take advantage of the negotiation model representation of an NESP to evaluate a binary optimization program at each time step to determine which, if any, negotiation actions should be initiated at that time step. Furthermore, if the agent classification map is unknown, agents are dynamically reclassified based on learning of the negotiation parameters exhibited by each agent in attempted negotiations.

4. **Empirical evaluation**:

In **Chapter 2**, we characterized the volatility of real hourly prices from the Ontario IESO wholesale electricity market to help understand the role of brokers in Smart Grid tariff markets. We determined that the expected impact of various weather-related factors such as temperature and wind speed are subsumed by the historical time series observations of the hourly prices. So, we developed an ARIMA time series model that generates forecasts when price changes fall within a “normal” range. A layered 3-label classification model predicts unusually large price changes outside of that range. Empirical analysis of these

wholesale prices and other tariff and customer data available from the current power grid and limited Smart Grid pilots provides a basis for further simulation experiments.

Also in **Chapter 2**, we evaluated the reinforcement learning strategy we developed for broker agents against various fixed and non-learning adaptive strategies and found that it almost always obtains the highest rewards. We showed that multiple broker agents using learned strategies each outperform non-learning broker agents. These results demonstrate that reinforcement learning presents an effective approach towards the development of autonomous broker agents for Smart Grid tariff markets. In **Chapter 3**, we evaluated the effectiveness of our augmented hierarchical Bayesian methodology by measuring the accuracy of forecasts within a given tolerance for errors and found that at 20% tolerance, our method is twice as accurate as traditional ARIMA forecasting. In **Chapter 4**, we demonstrated that our factored customer model can be used to instantiate a diverse set of customers. We also showed that our stochastic utility optimization algorithm achieves 5-12% cost savings for the members of the semi-cooperative neighborhood and simultaneously reduces the variance of their combined capacity levels without exhibiting *herding*, an undesirable peak-shifting behavior.

We contributed a non-monotonic exploration heuristic for *relearning* for dynamic environments with periodic changes. We demonstrated, using simulation-based experiments, that broker agents who use this relearning heuristic achieve higher rewards. We also contributed an analysis of the behaviors resulting from the interaction of multiple learning strategies in the tariff market. Specifically, we found that market prices are driven downwards rapidly and we found that the emergent aggregate broker agent rewards are largely consistent with economic principles, thus validating our simulation approach. We also analyzed real price data from a representative wholesale electricity market along with corresponding weather data to build a classification and forecasting model for hourly price changes.

In **Chapter 6**, we evaluated the Negotiated Learning technique, and the ATTRACTION-BOUNDED-LEARNING algorithm in particular, on the problems introduced in **Chapter 5**. Through experiments on the variable rate tariff selection problem, we demonstrated: (i) the value of negotiated information, (ii) the importance of a well-informed negotiation model, and (iii) learnability of negotiation models. We confirmed these findings using experiments on the capacity aggregate management problem and also studied the sensitivity to various algorithmic and environmental configuration parameters. We also studied the scalability of the ATTRACTION-BOUNDED-LEARNING with increasing numbers of semi-cooperative agents and demonstrated the value of agent classes in the NESP representation model.

8.2 Future Directions

This thesis presents new avenues for research on Smart Grid agents and multiagent learning in general. We briefly explore some such avenues in this section.

- We have already identified significant contributions by other researchers, *e.g.*, [Peters et al., 2013], which build upon our foundational work on Smart Grid broker agents. Our work focused on discretizing the continuous problem domain faced by such agents using domain-specific heuristics so that MDP-based reinforcement learning methods like Q-LEARNING can be applied effectively. Other reinforcement learning-based research efforts may seek to preserve the continuous nature of the state space and apply function approximation, evaluate the relative benefit of on-policy learning algorithms, or formulate alternate representations and apply other online learning methods for comparison.

- We primarily relied on ARIMA time series models as a basis for our analysis of wholesale prices and for simulation of customer capacity patterns. ARIMA models generally perform well when extensive historical time series are available for forecasting. However, we encountered simulation scenarios where the available histories are of limited length. Spectral learning algorithms and replication methods (*e.g.*, wavelets, predictive state representations [Singh et al., 2004]) may be evaluated for relative forecasting performance and for the ability to add subjective prior information into the forecasts.

- Additional factored models of Smart Grid customers may be instantiated at a finer granularity than we have done to evaluate how well the simulated behaviors scale down from population models to individual models. We hypothesize that fine-grained capacity originators can be used to model individual appliances, for example, to simulate the behavior of a household customer. A comparison of such a model with a fine-grained household customer model that explicitly models each appliance may present interesting paths for enhancing the versatility of the factored customer model framework.

- An immediate extension of Negotiable Entity Selection Processes would be derived by extending the bipartite graph in the negotiation model $\mathbf{N}$ to a bipartite multigraph; *i.e.*, allow multiple edges between the transformed states $\varphi(s(t), \mathcal{F})$ and the agent classes $\mathcal{K}$. Each of these edges connecting the same two nodes may carry different (c, τ, x) negotiation parameters representing tradeoffs in the cost, time required, and reliability of negotiations. Would such an extension be necessary for some problems? Would the added complexity be warranted in that problem domain?

- Applications of Negotiated Learning to problems of significantly larger scale may identify limitations that we have not considered. For example, what would happen if a problem introduced thousands of agent classes, thousands of negotiable entities, or hundreds of entity features. We hypothesize that the negotiation model and ATTRACTION-BOUNDED-LEARNING would simply yield a larger optimization problem to be solved and the algorithm would take longer to learn the agent classification maps or negotiation parameters, if they are hidden. Validating or refuting this hypothesis may be of benefit.

- We hope that other researchers will consider empirical evaluation of Negotiated Learning in the context of one of the problems identified in Section 5.4, or better yet, imagine and formulate other semi-cooperative problems beyond the Smart Grid domain that may be addressed by Negotiated Learning.

We conclude with some final discussion intended to aid future researchers in assessing the scope and applicability of the contributions of this thesis:

What if more data is available for some agents than other agents?
As an example, in the adaptive capacity optimization scenario, if some customers in the neighborhood have more data available to them than other customers, how does the asymmetry affect the semi-cooperative solution? This is a case where agent classes can be used to generalize information from a subset of agents to all agents in that class. So, for example, if we have hourly-metered capacity data for some customers but only daily-metered capacity data for other customers, we may generalize a mapping from daily capacity to hourly capacity patterns if agents in the class share other features such as home sizes, occupancy profiles, exposure to weather patterns, etc.

What if all data is freely available to all agents in the environment?
Within the Smart Grid domain, having all data freely available to all agents would in theory create a better solution because it would enable a central optimizer to evaluate the globally optimal solution for the entire ecosystem. However, this does not mean that each agent in the environment would be better off under such a global solution. If such an agent exists, that agent may be incentivized to hide some information to maximize self-interest, thus nullifying the assumption. Indeed, if we are to assume a sustainable competitive environment, then various agents offering competitive value propositions, *e.g.*, brokers offering tariffs, must find some method to distinguish themselves in the market. In the absence of such distinction, those agents would likely benefit from consolidation, thus eliminating competition in favor of monopolistic institutions.

When should one consider the application of Negotiated Learning to a problem?
In Section 5.4, we identified the criteria that define a suitable negotiable entity selection problem. However, considering that an algorithm based on exploration-exploitation tradeoffs can be applied to most such negotiable entity selection problems, there remains the question of when is Negotiated Learning likely to outperform an algorithm such as EXP3? The answer relates to *switching costs* and the dynamic nature of entity features. If switching frequently between entities is expensive and entity features evolve rapidly, Negotiated Learning allows for relatively inexpensive exploration and prediction of beneficial switching opportunities. When considering other domains, it is often useful to evaluate whether a switch involves physical infrastructure (*e.g.*, switching from cable TV to satellite TV) or only economic infrastructure. Switches involving physical infrastructure are often prohibitively expensive. Scenarios where switching costs are high due to significant economic costs of selecting the wrong entity at a particular time, and not simply due to high physical infrastructure switching costs, are ideal for Negotiated Learning.

When Negotiated Learning is applied to a different domain, what needs to change?
Data in different domains is likely to be intrinsically different in time scale and also on the contextual scale of what determines joint interests for the involved agents. For example, in the Smart Grid domain, we use hourly metering and therefore treat one hour as the unit for discrete time steps; however, in the *TV viewer* domain of Section 5.4, a five minute time step may be more appropriate. On the other hand, Smart Grid neighborhoods consist of geographically constrained agents compared to other TV viewers who may be across the country or the world. Moreover, the imputation and forecasting models used in ATTRACTION-BOUNDED-LEARNING have a strong dependence on domain-specific assumptions such as 24-hour periodicity or correlation with the hour-of-day. Alternate assumptions may be more justifiable and useful in other domains.

What is the expected impact of this thesis in the long run?
This thesis is driven by the belief that there exist exponentially more semi-cooperative interactions in the real world than there are fully cooperative or adversarial interactions. However, much prior work in multiagent systems has assumed rational agents in fully cooperative or adversarial environments. In our research, we have been inspired by Herb Simon's theory of *satisficing* and by Allen Newell's pursuit of *agents* as manifest artificial intelligence. We hope that this thesis—with its focus on semi-cooperative learning in heterogeneous agents of bounded rationality—facilitates the creation of intelligent computational agents that are more reflective of our human desire to maintain ultimate control over our actions and over the dissemination of our private data as we interact with an increasingly AI-enabled world.

Appendix A

Notation and Abbreviations

A.1 Guide to Notation

lowercase alphabet : a numeric variable (*e.g.*, price p), or an element of a set (*e.g.*, action a)

uppercase alphabet : a numeric constant (*e.g.*, horizon H), or a "thing" (*e.g.*, broker B)

script alphabet : a set (*e.g.*, set of actions $\mathcal{A}$)

bold alphabet : a model (*e.g.*, negotiation model $\mathbf{N}$)

uppercase roman : a function (*e.g.*, R for reward function)

blackboard bold : a number set (*e.g.*, real numbers $\mathbb{R}$), or the indicator function $\mathbb{1}$

italicized word : a categorical element of a set (*e.g.*, $\{$*Aggregator, Component*$\}$)

greek alphabet : a parameter (*e.g.*, threshold ξ), or intermediate value (*e.g.*, aggregate ζ)

A.2 List of Symbols

We have tried to use the above convention for notation throughout this document and have tried not to reuse symbols, except where necessitated by established convention (*e.g.*, e as exponent and e_t as the innovation at time t in ARIMA models), or where it is easily discernible from the context (*e.g.*, B for broker agent in Chapter 2 and for capacity bundle in Chapter 4).

Note that the following list omits a few symbols of minor significance such as those used for temporary variables in algorithms. It also omits subscripts and other denominations, which are introduced in context within the document. (See Section A.3 for a list of abbreviations.)

Symbol	**Description**	**Chapter(s)**
a	action in MDP/NESP	2, 5
$\mathcal{A}$	set of actions in MDP/NESP	2, 5
B	(i) broker	2
	(ii) capacity bundle in FCM	4
$\mathbf{B}$	neighbor state model in ABL	5
c	cost	*
C	consumer	*
D	distance between capacity profiles	4
e	innovation in ARIMA models, or exponent	*
f	entity feature in NESP	5
$\mathcal{F}$	set of entity features in NESP	5
h	labels for $\{-1, 0, 1\}$ classification	2
H	horizon for time series evaluation	*
$\mathcal{I}$	set of agents in NESP	5
j	index for iteration	*
k	agent class in NESP	5
$\mathcal{K}$	set of agent classes in NESP	5
$\mathbf{K}$	agent classification model in NESP	5
$\vec{L}$	profile recommendation (list of scored profiles)	4
$\mathcal{L}$	set of lookahead windows in ABL	5
m	anticipated value of negotiation action in ABL	5
$\mathcal{M}$	set of capacity originator models	4
$\vec{M}$	list of negotiation action values in ABL	5
n	negotiation instance in ABL	5
$\mathcal{N}$	set of ongoing negotiations in ABL	5
$\mathbf{N}$	negotiation model in NESP	5
o	capacity originator	*
p	price	*
P	producer	*
$\mathcal{P}$	set of capacity profiles	*
q	traded capacity in wholesale market	2
Q	(i) state-action value in Q-LEARNING	2
	(ii) component of capacity aggregate	5

Symbol	Description	Chapter(s)
r	reward in MDP/NESP	2, 5
R	reward function in MDP/NESP	2, 5
s	state in MDP/NESP	2, 5
S	tariff subscriber in FCM	4
$\mathcal{S}$	set of states in MDP	2
S	state model in NESP	5
t	time step	*
$\mathcal{T}$	time sequence	*
T	transition function in MDP/NESP	2, 5
u	utility value	4
U	utility function	4
U	utility optimizer in FCM	4
V	attraction (μ, β^+, β^-) in ABL	5
$\mathcal{V}$	set of attractions in ABL	5
w	weight	*
$\vec{W}$	array of 0-1 indicators in ABL	5
x	probability mass	*
$\mathcal{X}$	probability distribution	*
y	capacity (supply or demand) value	*
Y	capacity forecast	*
$\mathcal{Y}$	set of capacity forecasts	*
z	entity (*e.g.*, tariff) in NESP	5
$\mathcal{Z}$	set of entities in NESP	5
Z	simulated time series	3
$\mathbb{I}$	the set of all integers	*
$\mathbb{R}$	the set of all real numbers	*
α	learning rate in Q-LEARNING	2
β	attraction bounds $V.\beta^+$ and $V.\beta^-$ in ABL	5
γ	negotiation budget factor in ABL	5
Γ	imputation and forecasting methods in ABL	5
δ	price range for {-1,0,1} classification	2
ϵ	generically, a small value	*
ε	Gaussian noise	*

Symbol	**Description**	**Chapter(s)**
ζ	capacity aggregate	5
η	wholesale market clearing price	2
ϑ	lag time steps in temporal shifting profiles	4
θ	moving-average coefficient in ARIMA models	*
Θ	seasonal moving-average coefficient in ARIMA models	*
κ	customer capacity	2
λ	attractions bounds decay factor in ABL	5
Λ	decision-making agent	*
μ	(i) mean of a probability distribution (ii) attraction mean as in $V.\mu$ and (μ, β^+, β^-) in ABL	* 5
ν	ratio of consumers to producers	2
ξ	attraction benefit threshold in ABL	5
π	policy for a decision process	*
ρ	capacity profile	4, 5
σ	standard deviation of a probability distribution	*
ς	price increments in broker actions	2
τ	time period	*
υ	neighborhood utility score	4
φ	state transition in NESP	5
ϕ	autoregression coefficient in ARIMA models	*
ψ	negotiation budget in ABL	5
Φ	seasonal autoregression coefficient in ARIMA models	*
Ψ	broker's portfolio of customers	2
ω	attraction update weight in ABL	5
$\mathbb{1}$	0-1 indicator function	*

A.3 List of Abbreviations

ABL	ATTRACTION-BOUNDED-LEARNING
ACE	Agent-based Computational Economics
ACF	Autocorrelation Function
AMI	Advanced Metering Infrastructure
ARIMA	Autoregressive Integrated Moving Average
CPP	Critical Peak Pricing
DSM	Demand-side Management
DU	Distribution Utility
EWA	Experience Weighted Attraction
FCM	Factored Customer Model
FM	Forecasting Method
HBM	Hierarchical Bayesian Model
HOEP	Hourly Ontario Electricity Prices
HVAC	Heating, Ventilation and Air-Conditioning
IESO	Independent Electricity System Operator
IM	Imputation Method
LUMA	Learning Utility Management Agent
MDP	Markov Decision Process
NCDC	National Climatic Data Center
NESP	Negotiable Entity Selection Process
PACF	Partial Autocorrelation Function
PEV	Plugin Electric Vehicle
POMDP	Partially Observable Markov Decision Process
POSG	Partially Observable Stochastic Game
QRE	Quantal Response Equilibrium
RTP	Real-time Pricing
SARIMA	Seasonal Autoregressive Integrated Moving Average
SDGE	San Diego Gas & Electric
SVM	Support Vector Machine
TAC	Trading Agent Competition
TOU	Time of Use
VPP	Virtual Power Plant

Appendix B

Smart Grid Terminology

electricity / energy / power –

We generally use *energy*, measured in kWh, when referring to units of electricity that can be measured for billing and when referring to sources (*e.g.*, renewable energy resources, distributed energy sources). We use *power* usually when referring to the flow of electricity (*e.g.*, power production from energy sources, power transmission on the grid).

consumers / producers / customers –

Power systems literature refers to *loads* and *generators* to identify entities that consume and produce electricity respectively. We refer to them, using economic terms, as *consumers* and *producers*. Often we refer to the combined set of consumers and producers as *customers*.

capacity / demand / supply –

The term *capacity* is used traditionally to distinguish the potential of a load or generator to consume or produce power, as opposed to the realized power levels, which may be lower. We instead use *capacity* to refer uniformly to realized consumption and production levels; *i.e.*, *demand* and *supply*. We distinguish between potential and realized power levels using the economic notion of *elasticity*—the ratio of percentage change in demand or supply in response to a 1% change in an underlying causal variable.

power grid / Smart Grid –

The *power grid* is a network of electromechanical systems designed for the generation, transmission and distribution of electricity. Traditionally, it assumes a clean separation of *generation grids*, the *transmission grid*, and *distribution grids*, and also that demand is generally inelastic. *Smart Grid* is a loosely defined set of digital and economic control

systems intended to enhance the power grid to incorporate highly elastic demand, power generation within the distribution grid, and several other goals [Kannberg et al., 2003].

distribution utility / wholesale market –

Historically, electric utilities were responsible for both generation and distribution. Widespread deregulation has resulted in the separation of these responsibilities into distinct economic entities—*generating companies* and *distribution utilities*—who often trade in *wholesale markets* to buy/sell contracts for power supply.

broker / supplier / provider / aggregator –

In many locales, distribution utilities are monopolies that control both the physical infrastructure of the distribution grid and the supply of power to consumers over that grid. In some deregulated retail markets, the latter function is separated into a new role that is open to competition. Entities in the new role are variably called *suppliers*, *providers*, or *aggregators*. These terms do not sufficiently reflect an emerging responsibility of entities in this role—purchase of distributed generation from rooftop solar installations, etc. We instead use the term *broker* and formally define a broker's role in Chapter 2.

Appendix C

ARIMA Time Series Models

Autoregressive Integrated Moving Average (ARIMA) models are a family of stochastic process models typically used for one-dimensional time series analysis and forecasting.

- The simplest ARIMA time series model is for the **white noise** process:

$$Y_t = e_t \tag{C.1}$$

  where Y_t is the observation at time t and e_t is the **innovation** at t. Generally, innovations are estimated under a Gaussian assumption: $e_t \sim N(0, \sigma^2)$. When the equivalent model is used for forecasting, it is sometimes denoted as:

$$\hat{Y}_t \leftarrow N(0, \sigma^2) \tag{C.2}$$

  but we usually do not differentiate the notation for estimation and forecasting since the intention of the model is apparent from the context.

  The **order** of an ARIMA model is denoted $(\mathbf{p}, \mathbf{d}, \mathbf{q})$ where:

  - $\mathbf{p}$ is the order of the autoregressive components,
  - $\mathbf{d}$ is the order of *differencing*, explained below, and
  - $\mathbf{q}$ is the order of the moving average components.

  A Gaussian white noise process is identified as ARIMA$(0, 0, 0)$.

- The **grand mean** of a stationary process is:

$$\mu = Y_t - e_t \tag{C.3}$$

If μ is dependent on t, then the process is non-stationary. A **random walk** is defined as:

$$Y_t = Y_{t-1} + e_t \tag{C.4}$$

Moving terms around, we obtain a derived process called the **integrated** series, which can be used for estimating the model assuming stationarity:

$$X_t = Y_t - Y_{t-1} = e_t \tag{C.5}$$

This technique is called **differencing** and it is used to remove trend lines. It can be used repeatedly until a stationary process is obtained and the number of repetitions contributes $\mathbf{d}$ to the order of the ARIMA model.

- In a **moving average** (MA) process, the weighted impact of an innovation is reflected over multiple time steps. The number of such time steps contributes $\mathbf{q}$ to the order. So, an MA(1) process is an ARIMA$(0, 0, 1)$ process defined as:

$$Y_t = e_t + \theta_1 e_{t-1} \tag{C.6}$$

where $e_{t-1} = Y_{t-1} - Y_{t-2}$ and $\theta_1 \in \mathbb{R}$ weighs the impact of e_{t-1} on Y_t. Intuitively, we say that the process is *autocorrelated* for two time steps. A process that is autocorrelated for three time steps, ARIMA$(0, 0, 2)$, is defined as:

$$Y_t = e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} \tag{C.7}$$

where θ_2 is the weight of e_{t-2}. The model is thus extended to higher orders of $\mathbf{q}$. The **autocorrelation function** (ACF) is a diagnostic tool in identifying $\mathbf{q}$ for a given series. Figure 3.3 includes a plot of an example ACF.[1]

- An **autoregressive** (AR) process reflects the impact of an observation on *all* future observations. An exponential decay process, AR(1) or ARIMA$(1, 0, 0)$, is defined as:

$$Y_t = \phi_1 Y_{t-1} + e_t \tag{C.8}$$

where ϕ_1 of a well-behaved process is in $\mathbb{R}[-1, 1]$. $\phi_1 > 0$ generates smoothly decaying processes and $\phi_1 < 0$ generates an alternating decay process. An AR(2) process with

[1]The mechanics of the diagnosis methodology can be found in [Cryer and Chan, 2008].

appropriate values for ϕ_1 and ϕ_2 can model an observation's impact as a sinusoidal decay:

$$Y_t = \phi_2 Y_{t-2} + \phi_1 Y_{t-1} + e_t \tag{C.9}$$

The **partial autocorrelation function** (PACF) helps identify the autoregressive order $\mathbf{p}$.

- A **mixed** autoregressive moving average (ARMA) model simply combines the AR and MA components and estimates a single innovation for each t. For example, an ARMA$(2, 2)$ or ARIMA$(2, 0, 2)$ process is defined as:

$$Y_t = \phi_2 Y_{t-2} + \phi_1 Y_{t-1} + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} \tag{C.10}$$

- **Multiplicative seasonal ARIMA** (SARIMA) models extend ARIMA to include a multiplicative process at a lower frequency. The number of time steps in one seasonal period is the *seasonal order* **S**. Given hourly time steps, a daily cycle is modeled with $\mathbf{S} = 24$.

  A multiplicative seasonal model has order $(\mathbf{p}, \mathbf{d}, \mathbf{q}) \times (\mathbf{P}, \mathbf{D}, \mathbf{Q})_{\mathbf{S}}$. The methodology used to estimate $(\mathbf{P}, \mathbf{D}, \mathbf{Q})$ is similar to that for $(\mathbf{p}, \mathbf{d}, \mathbf{q})$. The seasonal autoregressive and moving average coefficients are denoted using Φ and Θ in correspondence with ϕ and θ.

  The moving average components for the two periodicities interact with each other, so their coefficients Θ and θ are multiplied together when weighing the innovation at the interaction points. This effect can be seen at $t - 25$ in the following SARIMA$(1, 0, 1) \times (1, 0, 1)_{24}$ process of Equation 3.1 where $\mu = Y_0$:

$$Y_t = Y_0 + \phi_1 Y_{t-1} + \Phi_1 Y_{t-24} + e_t + \theta_1 e_{t-1} + \Theta_1 e_{t-24} + \theta_1 \Theta_1 e_{t-25} \tag{C.11}$$

Appendix D

Power TAC Game Specification

The following is the abstract for the Power TAC 2013 Game Specification:

> This is the specification for the Power Trading Agent Competition for 2013 (Power TAC 2013). Power TAC is a competitive simulation that models a liberalized retail electrical energy market, where competing business entities or brokers offer energy services to customers through tariff contracts, and must then serve those customers by trading in a wholesale market. Brokers are challenged to maximize their profits by buying and selling energy in the wholesale and retail markets, subject to fixed costs and constraints. Costs include fees for publication and withdrawal of tariffs, and distribution fees for transporting energy to their contracted customers. Costs are also incurred whenever there is an imbalance between a brokers total contracted energy supply and demand within a given time slot.The simulation environment models a wholesale market, a regulated distribution utility, and a population of energy customers, situated in a real location on Earth during a specific period for which weather data is available. The wholesale market is a relatively simple call market, similar to many existing wholesale electric power markets, such as Nord Pool in Scandinavia or FERC markets in North America, but unlike the FERC markets we are modeling a single region, and therefore we do not model location-marginal pricing. Customer models include households and a variety of commercial and industrial entities, many of which have production capacity (such as solar panels or wind turbines) as well as electric vehicles. All have real-time metering to support allocation of their hourly supply and demand to their subscribed brokers, and all are approximate utility maximizers with respect to tariff selection, although the factors making up their utility functions may include aversion to change and complexity that can retard uptake of marginally better tariff offers. The distribution utility models the regulated natural monopoly that owns the regional distribution network, and is responsible for maintenance of its infrastructure and for real-time balancing of supply and demand. The balancing process is a market-based mechanism that uses economic incentives to encourage brokers to achieve balance within their portfolios of tariff subscribers and wholesale market positions, in the face of stochastic customer behaviors and weather-dependent renewable energy sources. The broker with the highest bank balance at the end of the simulation wins.

Broker Interactions

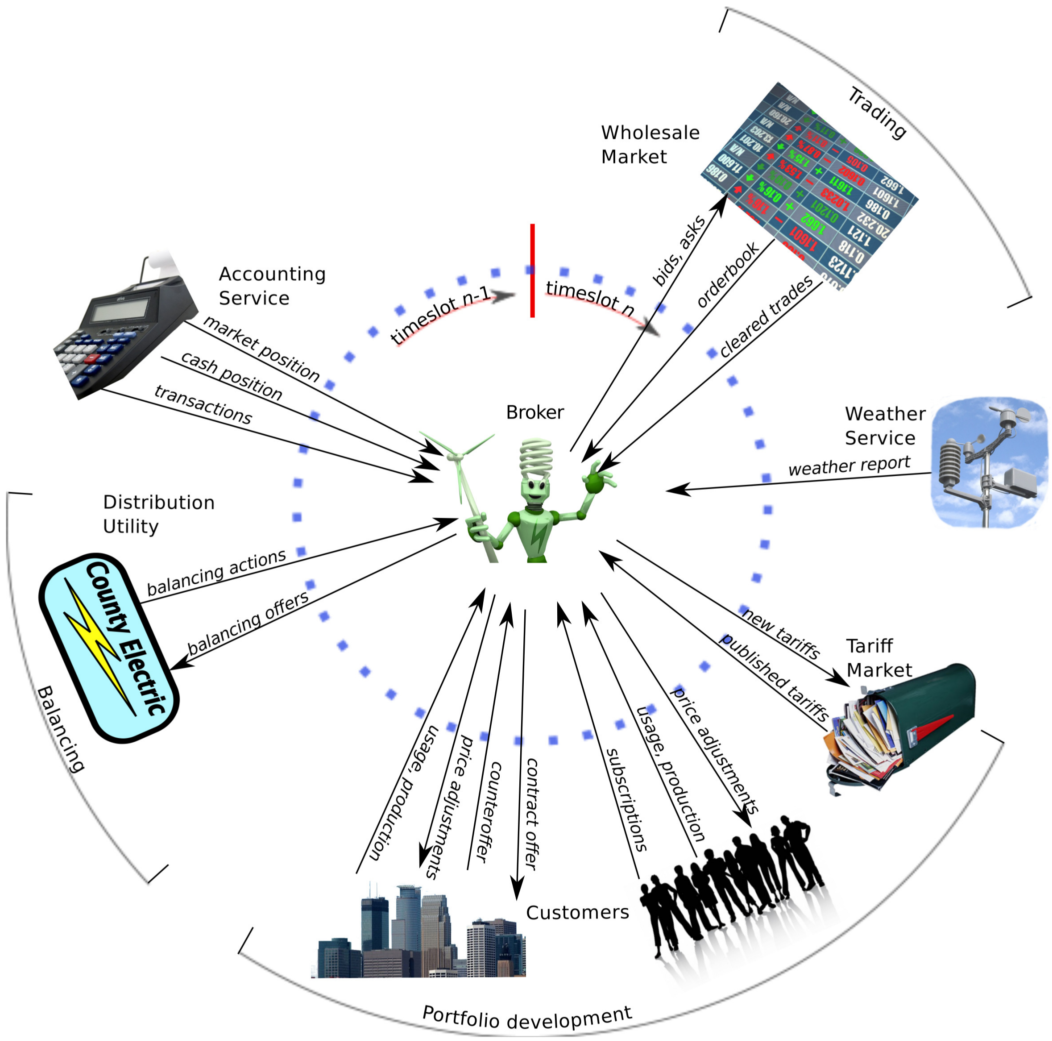


Figure D.1: Interactions of a broker agent with major components of the Power TAC environment. Each team participating in the competition develops one broker agent. [Ketter et al., 2013]

More Information

- The full game specification with the detailed rules and message definitions is available at:
  `http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2268852`
- More information about the Power TAC project and scheduled tournaments is available at:
  `http://www.powertac.org`

Appendix E

Tariff Ontology and Contracts

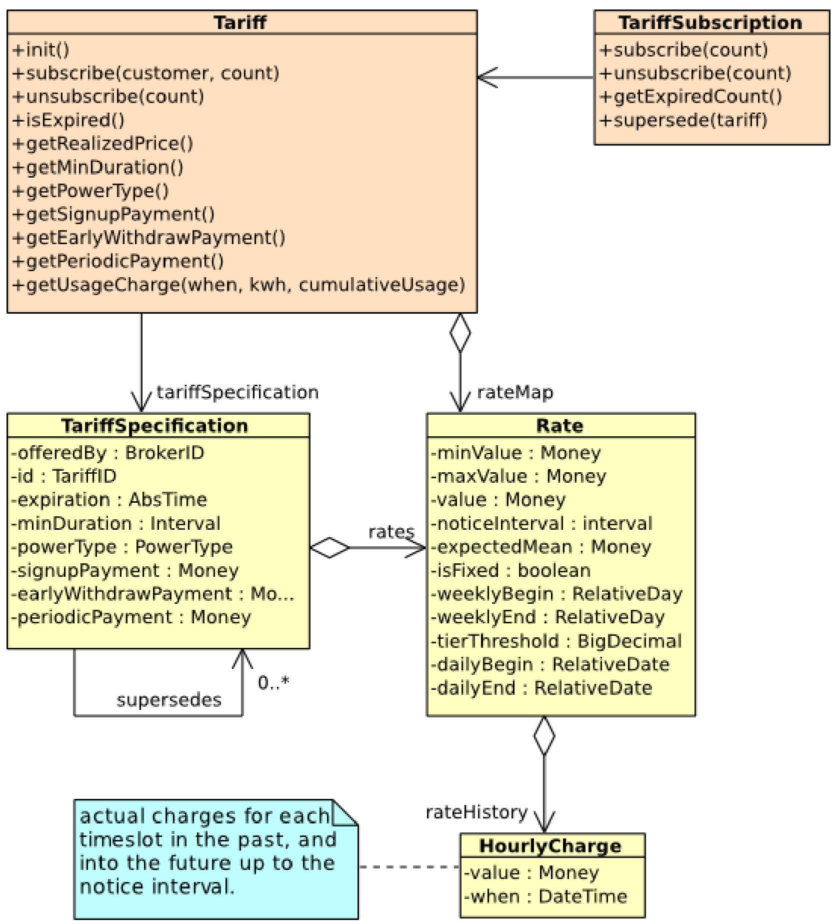


Figure E.1: Ontology for the structure of tariff contracts in the Power TAC simulation environment. The core terms of the contract are in the *TariffSpecification* and *Rate* entities while the *Tariff* and *TariffSubscription* entities provide layered behaviors and programming interfaces. *Source: http://powertac.org*

TariffSpecification Message Format

XML representation of a sample tariff specification from Power TAC:


```
<tariff-spec id="200000394"
        minDuration="0"
        powerType="WIND_PRODUCTION"
        signupPayment="0.0"
        earlyWithdrawPayment="0.0"
        periodicPayment="-1.0">
  <broker>Sample</broker>
  <rates>
    <rate id="200000395"
      tariffId="200000394"
      weeklyBegin="-1"
      weeklyEnd="-1"
      dailyBegin="-1"
      dailyEnd="-1"
      tierThreshold="0.0"
      isFixed="true"
      minValue="0.05345248201260664"
      maxValue="0.0"
      noticeInterval="0"
      expectedMean="0.0"
      maxCurtailment="0.0">
      <rateHistory/>
    </rate>
  </rates>
</tariff-spec>
```


Sample Tariff Contracts

Sample real world tariff contracts from deregulated tariff markets in the United States are available at: `http://www.cs.cu.edu/~ppr/thesis`

Appendix F

Factored Customer Instances

The following configuration was used to generate the capacity patterns of Figure 4.4.

`http://www.cs.cmu.edu/~ppr/thesis/plus/fcm-config.xml`

Note:

- The configuration includes six factored customer instantiations with eight capacity bundles.
- The *WindmillCoop* customer has two capacity bundles with differing characteristics.
- The *MedicalCenter* customer has one consumption bundle and one production bundle.
- The *BrooksideHomes* and *CentervilleHomes* customers are largely similar but vary in some respects as described in the embedded comments, demonstrating how customers can be configured with variations to test hypotheses.

```xml
<customers>

   <customer name="BrooksideHomes" count="1" creatorKey="" entityType="RESIDENTIAL">
      <description>
      A prototypical multicontracting consumer population of 30000 households in a suburban area
       with consumption peaking in the morning and evening. The aggregate capacity is generated
       using a timeseries forecasting model learned from a fine-grained model based on the MEREGIO
       project household model developed by Gottwalt, et al.
      </description>
      <capacityBundle id="" population="30000" powerType="CONSUMPTION" multiContracting="true" canNegotiate="false">
         <tariffSubscriber>
            <constraints>
               <benchmarkRisk enable="true" ratio="10:1" />
               <tariffThrottling enable="true" />
            </constraints>
            <influenceFactors>
               <priceWeights expMean="0.6" maxValue="0.4" realized="0.75" />
            </influenceFactors>
            <allocation method="LOGIT_CHOICE">
               <logitChoice rationality="0.9" />
            </allocation>
            <reconsideration period="28"/>
            <switchingInertia>
               <inertiaDistribution distribution="INTERVAL" mean="0.3" stdDev="0.1" low="0" high="1" />
            </switchingInertia>
         </tariffSubscriber>
         <capacity count="1" description="Collection of suburban households.">
```

```
        <baseCapacity type="TIMESERIES">
          <timeseriesModel type="ARIMA_101x101">
            <modelParams name="data/BrooksideHomesModelParams.dat" source="CLASSPATH" />
            <refSeries name="data/BrooksideHomesRefSeries.dat" source="CLASSPATH" />
          </timeseriesModel>
        </baseCapacity>
        <influenceFactors>
          <dailySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
          <hourlySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 0.5, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 0.5,
              1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
          <temperature influence="DEVIATION" reference="20" rangeMap="-50~-21: +0.005, -20~0: +0.01, 1~16: +0.02,
              17~24: 0.00, 25~35: +0.01, 36~50: +0.005" />
          <windSpeed influence="NONE" rangeMap="" />
          <windDirection influence="NONE" rangeMap="0~360: 1.0" />
          <cloudCover influence="NONE" />
          <priceElasticity>
            <benchmarkRates rangeMap="00~23: -0.15" />
            <elasticityModel type="CONTINUOUS" ratio="-0.01" range="0.7~1.0" />
          </priceElasticity>
        </influenceFactors>
      </capacity>
    </capacityBundle>
  </customer>

  <customer name="CentervilleHomes" count="1" creatorKey="LEARNING" entityType="RESIDENTIAL">
    <description>
    A multicontracting consumer population of 20000 households that is similar to BrooksideHomes in many aspects.
    There are three key distinctions: (i) the capacities for these customers are generated using a population
    distribution model instead of a timeseries forecasting model, (ii) the capacities exhibit decision-theoretic
    adaptive behavior whereby they adjust the time-shifting of capacities dynamically when presented with TOU tariffs,
    and (iii) tariffs are reevaluated more frequently.
    </description>
    <capacityBundle id="" population="20000" powerType="CONSUMPTION" multiContracting="true" canNegotiate="false">
      <tariffSubscriber>
        <constraints>
          <benchmarkRisk enable="true" ratio="10:1" />
          <tariffThrottling enable="true" />
        </constraints>
        <influenceFactors>
          <priceWeights expMean="0.6" maxValue="0.4" realized="0.95" />
        </influenceFactors>
        <allocation method="LOGIT_CHOICE">
          <logitChoice rationality="1.0" />
        </allocation>
        <reconsideration period="8"/>
        <switchingInertia>
          <inertiaDistribution distribution="INTERVAL" mean="0.3" stdDev="0.1" low="0" high="1" />
        </switchingInertia>
      </tariffSubscriber>
      <capacity count="1" description="Collection of urban households.">
        <baseCapacity type="POPULATION">
          <populationCapacity distribution="NORMAL" mean="22000" stdDev="2000" />
        </baseCapacity>
        <influenceFactors>
          <dailySkew array="0.8, 0.8, 0.8, 0.8, 0.8, 1.0, 0.9" />
          <hourlySkew array="0.5, 0.4, 0.4, 0.5, 0.5, 0.6, 0.6, 0.6, 0.7, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.7,
              0.8, 0.9, 1.0, 1.0, 0.9, 0.7, 0.5" />
          <temperature influence="DEVIATION" reference="20" rangeMap="-50~-21: +0.005, -20~0: +0.01, 1~16: +0.02,
              17~24: 0.00, 25~35: +0.01, 36~50: +0.005" />
          <windSpeed influence="NONE" rangeMap="" />
          <windDirection influence="NONE" rangeMap="0~360: 1.0" />
          <cloudCover influence="NONE" />
          <priceElasticity>
            <benchmarkRates rangeMap="00~23: -0.15" />
            <elasticityModel type="CONTINUOUS" ratio="-0.01" range="0.7~1.0"/>
          </priceElasticity>
        </influenceFactors>
      </capacity>
    </capacityBundle>
  </customer>
```

```xml
<customer name="DowntownOffices" count="1" creatorKey="" entityType="COMMERCIAL">
    <description>
    A multicontracting consumer population of 30 urban offices that are similar to household populations
    in most aspects except that the consumption pattern is consistently sustained at a higher level
    during weekdays and is lower at night and on the weekends.
    </description>
    <capacityBundle id="" population="30" powerType="CONSUMPTION" multiContracting="true" canNegotiate="false">
        <tariffSubscriber>
            <constraints>
                <benchmarkRisk enable="true" ratio="10:1" />
                <tariffThrottling enable="true" />
            </constraints>
            <influenceFactors>
                <priceWeights expMean="0.6" maxValue="0.4" realized="0.8" />
            </influenceFactors>
            <allocation method="LOGIT_CHOICE">
                <logitChoice rationality="0.9" />
            </allocation>
            <reconsideration period="8"/>
            <switchingInertia>
                <inertiaDistribution distribution="INTERVAL" mean="0.3" stdDev="0.1" low="0" high="1" />
            </switchingInertia>
        </tariffSubscriber>
        <capacity count="1" description="Downtown office buildings.">
            <baseCapacity type="POPULATION">
                <populationCapacity distribution="NORMAL" mean="8000" stdDev="500" />
            </baseCapacity>
            <influenceFactors>
                <dailySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 0.6, 0.6" />
                <hourlySkew array="0.3, 0.3, 0.3, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
                      0.9, 0.7, 0.5, 0.4, 0.3, 0.3, 0.3" />
                <temperature influence="DEVIATION" reference="20" rangeMap="-50~-21: +0.005, -20~0: +0.01, 1~16: +0.02,
                      17~24: 0.00, 25~35: +0.01, 36~50: +0.005" />
                <windSpeed influence="NONE" />
                <windDirection influence="NONE" />
                <cloudCover influence="NONE" />
                <priceElasticity>
                    <benchmarkRates rangeMap="00~05: -0.10, 06~19: -0.20, 20~23: -0.10" />
                    <elasticityModel type="STEPWISE" map="1.5: 0.9, 2.0: 0.8" />
                </priceElasticity>
            </influenceFactors>
        </capacity>
    </capacityBundle>
</customer>

<customer name="MedicalCenter" count="1" creatorKey="" entityType="COMMERCIAL">
    <description>
    A hybrid customer representing a single large hospital complex with a large consumption
    capacity and a small solar production capacity. The consumption and production capacities
    may be allocated to tariffs from different brokers. Consumption is skewed towards slightly
    lower consumption at night and notably lower over the weekend, although much higher than
    is typical for commercial office buildings. Production capacity is mostly governed by
    daylight and cloud cover.
    </description>
    <capacityBundle id="1" population="1" powerType="CONSUMPTION" multiContracting="false" canNegotiate="false">
        <tariffSubscriber>
            <constraints>
                <benchmarkRisk enable="true" ratio="10:1" />
                <tariffThrottling enable="true" />
            </constraints>
            <influenceFactors>
                <priceWeights expMean="0.6" maxValue="0.4" realized="0.75" />
            </influenceFactors>
            <allocation method="LOGIT_CHOICE">
                <logitChoice rationality="0.9" />
            </allocation>
            <reconsideration period="16"/>
            <switchingInertia>
                <inertiaDistribution distribution="INTERVAL" mean="0.3" stdDev="0.1" low="0" high="1" />
            </switchingInertia>
        </tariffSubscriber>
```

```xml
      <capacity count="1" description="Facilities in hospital complex.">
        <baseCapacity type="POPULATION">
          <populationCapacity distribution="NORMAL" mean="5000" stdDev="500" />
        </baseCapacity>
        <influenceFactors>
          <dailySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 0.9, 0.9" />
          <hourlySkew array="0.7, 0.7, 0.7, 0.7, 0.7, 0.8, 0.9, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
               1.0, 1.0, 0.9, 0.8, 0.7, 0.7, 0.7" />
          <temperature influence="DEVIATION" reference="20" rangeMap="-50~-21: +0.005, -20~0: +0.01, 1~16: +0.02,
               17~24: 0.00, 25~35: +0.01, 36~50: +0.005" />
          <windSpeed influence="NONE" />
          <windDirection influence="NONE" />
          <cloudCover influence="NONE" />
          <priceElasticity>
            <benchmarkRates rangeMap="00~05: -0.10, 06~19: -0.20, 20~23: -0.10" />
            <elasticityModel type="STEPWISE" map="1.3: 0.8, 1.5: 0.7" />
          </priceElasticity>
        </influenceFactors>
      </capacity>
    </capacityBundle>
    <capacityBundle id="2" population="1" powerType="SOLAR_PRODUCTION" multiContracting="false" canNegotiate="false">
      <tariffSubscriber>
        <constraints>
          <benchmarkRisk enable="true" ratio="10:1" />
          <tariffThrottling enable="true" />
        </constraints>
        <influenceFactors>
          <priceWeights expMean="0.6" maxValue="0.4" realized="0.6" />
        </influenceFactors>
        <allocation method="LOGIT_CHOICE">
          <logitChoice rationality="0.95" />
        </allocation>
        <reconsideration period="8"/>
        <switchingInertia>
          <inertiaDistribution distribution="POINTMASS" value="0.5" />
        </switchingInertia>
      </tariffSubscriber>
      <capacity count="1" description="Solar capacity in hospital complex.">
        <baseCapacity type="INDIVIDUAL">
          <individualCapacity distribution="NORMAL" mean="1000" stdDev="50" />
        </baseCapacity>
        <influenceFactors>
          <dailySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
          <hourlySkew array="0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
               0.5, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0" />
          <temperature influence="NONE" />
          <windSpeed influence="NONE" />
          <windDirection influence="NONE" />
          <cloudCover influence="DIRECT" percentMap="0~40: 1.0, 41~60: 0.95, 61~75: 0.9, 76~90: 0.8, 91~100: 0.7" />
          <priceElasticity>
            <benchmarkRates rangeMap="00~23: 0.08" />
            <elasticityModel type="CONTINUOUS" ratio="-0.0001" range="0.9~1.0"/>
          </priceElasticity>
        </influenceFactors>
      </capacity>
    </capacityBundle>
  </customer>

  <customer name="SunnyhillSolar" count="1" creatorKey="" entityType="INDUSTRIAL">
    <description>
    A small community-owned PEV farm that is managed as a single unit and therefore
    has no multicontracting capabilities. The generated capacity is driven largely
    by daylight and cloud cover.
    </description>
    <capacityBundle id="" population="1" powerType="SOLAR_PRODUCTION" multiContracting="false" canNegotiate="false">
      <tariffSubscriber>
        <constraints>
          <benchmarkRisk enable="true" ratio="10:1" />
          <tariffThrottling enable="true" />
        </constraints>
        <influenceFactors>
```

```
        <priceWeights expMean="0.6" maxValue="0.4" realized="0.8" />
      </influenceFactors>
      <allocation method="LOGIT_CHOICE">
        <logitChoice rationality="0.95" />
      </allocation>
      <reconsideration period="8"/>
      <switchingInertia>
        <inertiaDistribution distribution="POINTMASS" value="0.1" />
      </switchingInertia>
    </tariffSubscriber>
    <capacity count="1" description="Community solar farm.">
      <baseCapacity type="INDIVIDUAL">
        <individualCapacity distribution="NORMAL" mean="6000" stdDev="300" />
      </baseCapacity>
      <influenceFactors>
        <dailySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
        <hourlySkew array="0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
            0.8, 0.8, 0.6, 0.4, 0.0, 0.0, 0.0" />
        <temperature influence="NONE" />
        <windSpeed influence="NONE" />
        <windDirection influence="NONE" />
        <cloudCover influence="DIRECT" percentMap="0~30: 1.0, 31~50: 1.1, 51~70: 1.0, 71~80: 0.9, 81~90: 0.8,
            91~100: 0.7" />
        <priceElasticity>
          <benchmarkRates rangeMap="00~23: 0.08" />
          <elasticityModel type="CONTINUOUS" ratio="-0.001" range="0.8~1.0"/>
        </priceElasticity>
      </influenceFactors>
    </capacity>
  </capacityBundle>
</customer>

<customer name="WindmillCoOp" count="1" creatorKey="" entityType="INDUSTRIAL">
  <description>
  A cooperative of 90 rural wind turbines with muticontracting production. 50 of the
  wind turbines are placed to maximize production from south-westerly winds and 40 are
  placed to maximize production from south-easterly winds.
  </description>
  <capacityBundle id="1" population="50" powerType="WIND_PRODUCTION" multiContracting="true" canNegotiate="false">
    <tariffSubscriber>
      <constraints>
        <benchmarkRisk enable="true" ratio="10:1" />
        <tariffThrottling enable="true" />
      </constraints>
      <influenceFactors>
        <priceWeights expMean="0.6" maxValue="0.4" realized="0.8" />
      </influenceFactors>
      <allocation method="LOGIT_CHOICE">
        <logitChoice rationality="0.95" />
      </allocation>
      <reconsideration period="28"/>
      <switchingInertia>
        <inertiaDistribution distribution="INTERVAL" mean="0.2" stdDev="0.1" low="0" high="1" />
      </switchingInertia>
    </tariffSubscriber>
    <capacity count="1" description="First subset of small windmills.">
      <baseCapacity type="INDIVIDUAL">
        <individualCapacity distribution="NORMAL" mean="100" stdDev="20" />
      </baseCapacity>
      <influenceFactors>
        <dailySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
        <hourlySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
            1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
        <temperature influence="NONE" />
  <windSpeed influence="DIRECT" rangeMap="0~3: 0.0, 3~6: 0.5, 7~17: 1.0, 18~30: 1.2, 31~105: 0.0" />
        <windDirection influence="DIRECT" rangeMap="0~90: 0.5, 91~180: 0.8, 181~270: 1.0, 271~360: 0.9" />
        <cloudCover influence="NONE" />
        <priceElasticity>
          <benchmarkRates rangeMap="00~23: 0.08" />
          <elasticityModel type="STEPWISE" map="0.5: 0.8, 0.75: 0.9" />
        </priceElasticity>
```

```xml
            </influenceFactors>
        </capacity>
    </capacityBundle>
    <capacityBundle id="2" population="40" powerType="WIND_PRODUCTION" multiContracting="true" canNegotiate="false">
        <tariffSubscriber>
            <constraints>
                <benchmarkRisk enable="true" ratio="10:1" />
                <tariffThrottling enable="true" />
            </constraints>
            <influenceFactors>
                <priceWeights expMean="0.6" maxValue="0.4" realized="0.8" />
            </influenceFactors>
            <allocation method="LOGIT_CHOICE">
                <logitChoice rationality="1.0" />
            </allocation>
            <reconsideration period="4"/>
            <switchingInertia>
                <inertiaDistribution distribution="POINTMASS" value="0.0" />
            </switchingInertia>
        </tariffSubscriber>
        <capacity count="1" description="Second subset of small windmills.">
            <baseCapacity type="INDIVIDUAL">
                <individualCapacity distribution="NORMAL" mean="180" stdDev="15" />
            </baseCapacity>
            <influenceFactors>
                <dailySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
                <hourlySkew array="1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
                      1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0" />
                <temperature influence="NONE" />
                <windSpeed influence="DIRECT" rangeMap="0~2: 0.0, 3~4: 0.4, 5~6: 0.7, 7~17: 1.0, 18~30: 1.2, 31~105: 0.0" /
                      >
                <windDirection influence="DIRECT" rangeMap="0~90: 0.8, 91~180: 1.0, 181~270: 0.9, 271~360: 0.5" />
                <cloudCover influence="NONE" />
                <priceElasticity>
                    <benchmarkRates rangeMap="00~23: 0.08" />
                    <elasticityModel type="STEPWISE" map="0.5: 0.8, 0.75: 0.9" />
                </priceElasticity>
            </influenceFactors>
        </capacity>
    </capacityBundle>
</customer>

</customers>
```

Bibliography

[Allen et al., 2001] Allen, E., LaWhite, N., Yoon, Y., Chapman, J., and Ilic, M. (2001). Interactive Object-Oriented Simulation of Interconnected Power Systems using SIMULINK. *IEEE Transactions on Education, vol 44*.

[Amin and Wollenberg, 2005] Amin, M. and Wollenberg, B. (2005). Toward a smart grid: Power delivery for the 21st century. *IEEE Power & Energy*, 3(5):34–41.

[Armstrong et al., 2009] Armstrong, M., Swinton, M., Ribberink, H., Beausoleil-Morrison, I., and Millette, J. (2009). Synthetically derived profiles for representing occupant-driven electric loads in Canadian housing. *Journal of Building Performance Simulation*, 2:1530.

[Armstrong-Crews and Veloso, 2007] Armstrong-Crews, N. and Veloso, M. (2007). Oracular Partially Observable Markov Decision Processes: A Very Special Case. In *Proceedings of the IEEE International Conference on Robotics and Automation*.

[Armstrong-Crews and Veloso, 2008] Armstrong-Crews, N. and Veloso, M. (2008). An approximate algorithm for solving oracular POMDPs. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

[Auer et al., 1995] Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (1995). Gambling in a rigged casino: The adversarial multi-armed bandit problem. *Proceedings of IEEE 36th Annual Foundations of Computer Science*, 68(68):322–331.

[Auer et al., 2002] Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (2002). The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77.

[Bajcsy, 1988] Bajcsy, R. (1988). Active perception. *Proceedings of the IEEE*, 76(8):966–1005.

[Barber, 2007] Barber, K. (2007). Multi-Scale Behavioral Modeling and Analysis Promoting a Fundamental Understanding of Agent-Based System Design and Operation. Technical report, DARPA Final Report by University of Texas, Austin (AFRL-IF-RS-TR-2007-58).

[Barbose et al., 2005] Barbose, G., Goldman, C., and Neenan, B. (2005). Electricity in real time–a survey of utility experience with real time pricing. *Energy*, 30.

[Berliner, 1995] Berliner, L. M. (1995). Hierarchical Bayesian Time Series Models. In *Workshop on Maximum Entropy and Bayesian Methods*.

[Blackhurst et al., 2011] Blackhurst, M., Lima Azevedo, I., Scott Matthews, H., and Hendrickson, C. T. (2011). Designing building energy efficiency programs for greenhouse gas reductions. *Energy Policy*, 39(9):5269–5279.

[Block et al., 2010] Block, C., Collins, J., and Ketter, W. (2010). A Multi-Agent Energy Trading Competition. Technical report, Erasmus University Rotterdam.

[Blum and Mansour, 2007] Blum, A. and Mansour, Y. (2007). From External to Internal Regret. *Journal of Machine Learning Research*, 8(1079):1307–1324.

[Boots and Gordon, 2011] Boots, B. and Gordon, G. J. (2011). Predictive State Temporal Difference Learning. In *Proceedings of Advances in Neural Information Processing Systems 24*.

[Borenstein, 2002] Borenstein, S. (2002). The Trouble With Electricity Markets: Understanding California's Restructuring Disaster. *Journal of Economic Perspectives*.

[Boutilier, 1996] Boutilier, C. (1996). Planning, learning and coordination in multiagent decision processes. In *Proceedings of the 6th conference on Theoretical aspects of rationality and knowledge*, pages 195–210. Morgan Kaufmann Publishers Inc.

[Bowling and Veloso, 2003] Bowling, M. and Veloso, M. (2003). Simultaneous Adversarial Multi-Robot Learning. In *International Joint Conference on Artificial Intelligence (IJCAI)*.

[Boyan, 1999] Boyan, J. (1999). Least-Squares Temporal Difference Learning. *Science*, 49:49–56.

[Bradtke and Barto, 1996] Bradtke, S. J. and Barto, A. G. (1996). Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22(1-3):33–57.

[Brafman and Domshlak, 2008] Brafman, R. and Domshlak, C. (2008). Preference Handling - An Introductory Tutorial. Technical report, TR 08-04, Ben-Gurion University.

[Brafman and Tennenholtz, 2002] Brafman, R. I. and Tennenholtz, M. (2002). R-MAX – A General Polynomial Time Algorithm for Near-Optimal Reinforcement Learning. *Journal of Machine Learning Research*, 3:213–231.

[Braun and Strauss, 2008] Braun, M. and Strauss, P. (2008). Aggregation approaches of controllable distributed energy units in electrical power systems. *Journal of Distributed Energy Resources*.

[Brown, 1951] Brown, G. (1951). Iterative solution of games by fictitious play. *Activity Analysis of Production and Allocation*.

[Camerer, 2003] Camerer, C. F. (2003). *Behavioral Game Theory*. Princeton University Press.

[Camerer, 2008] Camerer, C. F. (2008). Behavioral Game Theory and the Neural Basis of Strategic Choice. In *Neuroeconomics: Formal Models Of Decision-Making And Cognitive Neuroscience*, pages 193–206. Academic Press.

[Camerer and Ho, 1999] Camerer, C. F. and Ho, T. H. (1999). Experience-Weighted Attraction Learning in Normal Form Games. *Econometrica*, 67(4):827–874.

[Chalkiadakis et al., 2011] Chalkiadakis, G., Robu, V., Kota, R., Rogers, A., and Jennings, N. R. (2011). Cooperatives of Distributed Energy Resources for Efficient Virtual Power Plants. In *Autonomous Agents and Multi-Agent Systems*.

[Chernova and Veloso, 2009] Chernova, S. and Veloso, M. (2009). Interactive Policy Learning through Confidence-Based Autonomy. *Journal of Artificial Intelligence Research*, 34(1):1–25.

[Cheung and Friedman, 1997] Cheung, Y.-W. and Friedman, D. (1997). Individual learning in normal form games: Some laboratory results. *Games and Economic Behavior*, 19:46–76.

[Chong et al., 2006] Chong, J.-K., Camerer, C. F., and Ho, T. H. (2006). A Learning-based Model of Repeated Games with Incomplete Information. *Games and Economic Behavior*, 55(2):340–371.

[Chrysopoulos and Symeonidis, 2009] Chrysopoulos, A. and Symeonidis, A. (2009). Improving Agent Bidding in Power Stock Markets Through A Data Mining Enhanced Agent Platform. In *Agents and Data Mining Interaction workshop, AAMAS'09*.

[Conitzer and Sandholm, 2006] Conitzer, V. and Sandholm, T. (2006). AWESOME: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents. *Machine Learning*, 67(1-2):23–43.

[Contreras et al., 2001] Contreras, J., Candiles, O., de la Fuente, J., and Gomez, T. (2001). Auction design in day-ahead electricity markets. *IEEE Transactions in Power Systems, 16(3)*.

[Cournot, 1838] Cournot, A. (1838). *Researches in the Mathematical Principles of the Theory of Wealth*. Haffner.

[Crawford and Veloso, 2008] Crawford, E. and Veloso, M. (2008). Negotiation in Semi-Cooperative Agreement Problems. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*.

[Cryer and Chan, 2008] Cryer, J. and Chan, K. (2008). *Time Series Analysis with Applications in R*. Springer.

[David and Wen, 2000] David, A. and Wen, F. (2000). Strategic bidding in competitive electricity markets: a literature survey. In *IEEE Power Engineering Society*.

[Dietterich, 1999] Dietterich, T. G. (1999). Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition. *Journal of Artificial Intelligence Research*, 13(1):63.

[DoE, 2010] DoE (2010). http://www.eia.doe.gov.

[Dolbow et al., 2004] Dolbow, J., Khaleel, M. A., and Mitchell, J. (2004). Multiscale Mathematics Initiative: A Roadmap. Technical Report December, Pacific Northwest National Laboratory.

[Doshi et al., 2008] Doshi, P., Zeng, Y., and Chen, Q. (2008). Graphical models for interactive POMDPs: representations and solutions. *Autonomous Agents and MultiAgent Systems*, 18(3):376–416.

[Erev and Roth, 1998] Erev, I. and Roth, A. E. (1998). Predicting how people play games: Reinforcement learning in experimental games with unique, mized-strategy equilibria. *American Economic Review*.

[Faruqui and Palmer, 2012] Faruqui, A. and Palmer, J. (2012). Dynamic Pricing and Its Discontents. Technical report, The Brattle Group.

[Foster and Vohra, 1999] Foster, D. P. and Vohra, R. (1999). Regret in the On-Line Decision Problem. *Games and Economic Behavior*, 29(1-2):7–35.

[Fudenberg and Levine, 1995] Fudenberg, D. and Levine, D. (1995). Universal consistency and cautious fictitious play. *Journal of Economic Dynamics and Control*.

[Gellings et al., 2004] Gellings, C., Samotyj, M., and Howe, B. (2004). The future power delivery system. *IEEE Power & Energy*, 2(5):40–48.

[Gelman and Hill, 2007] Gelman, A. and Hill, J. (2007). *Data Analysis Using Regression and Multilevel/Hierarchical Models.* Cambridge University Press.

[Geman and Geman, 1984] Geman, S. and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence 6 (6): 721741*.

[Gmytrasiewicz and Doshi, 2005] Gmytrasiewicz, P. and Doshi, P. (2005). A Framework for Sequential Planning in Multi-Agent Settings. *Journal of Artificial Intelligence Research (JAIR)*, 24:49–79.

[Gomes, 2009] Gomes, C. (2009). Computational Sustainability: Computational Methods for a Sustainable Environment. *The Bridge, National Academy of Engineering*, 39.

[Gottwalt et al., 2011] Gottwalt, S., Ketter, W., Block, C., Collins, J., and Weinhardt, C. (2011). Demand side management - a simulation of household behavior under variable prices. *Energy Policy*, 39:8163–8174.

[Greenwald and Hall, 2003] Greenwald, A. and Hall, K. (2003). Correlated-Q learning. *Machine Learning*, pages 242–249.

[Groves, 1973] Groves, T. (1973). Incentives in Teams. *Econometrica*, 41(4):617–631.

[Guestrin et al., 2003] Guestrin, C., Koller, D., Parr, R., and Venkataraman, S. (2003). Efficient Solution Algorithms for Factored MDPs. In *Journal of Artificial Intelligence Research (JAIR)*, volume 19, pages 399–468.

[Guo et al., 2008] Guo, Y., Li, R., Poulton, G., and Zeman, A. (2008). A Simulator for Self-Adaptive Energy Demand Management. In *IEEE Conf. on Self-Adaptive and Self-Organizing Systems*.

[Hammerstrom, 2008] Hammerstrom, D. (2008). Pacific Northwest GridWise Testbed Demonstration Projects; Part I. Olympic Peninsula Project. Technical report, PNNL-17167, Pacific Northwest National Laboratory.

[Hannan, 1957] Hannan, J. (1957). Approximation to Bayes risk in repeated plays. *Theory of Games*.

[Hansen et al., 2004] Hansen, E. A., Bernstein, D. S., and Zilberstein, S. (2004). Dynamic Programming for Partially Observable Stochastic Games. *Artificial Intelligence*, 9(2000):709–715.

[Hart, 2008] Hart, D. (2008). Using AMI to realize the Smart Grid. In *IEEE Power Engineering Society*.

[Hart and Mas-Colell, 2000] Hart, S. and Mas-Colell, A. (2000). A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68:1127–1150.

[Hirsch et al., 2010] Hirsch, C., Hillemacher, L., Block, C., and Schuller, A. (2010). Simulations in the smart grid field study MeRegio. *Information Technology*, 52:100106.

[Horvitz et al., 1988] Horvitz, E., Breese, J., and Henrion, M. (1988). Decision theory in expert systems and artificial intelligence. *Int. Journal of Approximate Reasoning, 2, 247-302*.

[Howard, 1966] Howard, R. A. (1966). Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, SSC-2:22–26.

[Howard and Matheson, 1984] Howard, R. A. and Matheson, J. E. (1984). Influence Diagrams. *Principles and Applications of Decision Analysis*.

[Hu and Wellman, 1998] Hu, J. and Wellman, M. P. (1998). Multiagent reinforcement learning: Theoretical framework and an algorithm. *Proceedings of the Fifteenth International Conference on Machine Learning*.

[Hu and Wellman, 2003] Hu, J. and Wellman, M. P. (2003). Nash Q-Learning for General-Sum Stochastic Games. *Journal of Machine Learning Research*, 4(6):1039–1069.

[IESO, 2011] IESO (2011). http://www.ieso.ca.

[Jones et al., 2003] Jones, R., Ghani, R., Mitchell, T., and Riloff, E. (2003). Active Learning with Multiple View Feature Sets. In *ECML 2003 Workshop on Adaptive Text Extraction and Mining*.

[Joskow and Tirole, 2006] Joskow, P. and Tirole, J. (2006). Retail electricity competition. *The Rand Journal of Economics*, 37(4):799–815.

[Joskow, 2008] Joskow, P. L. (2008). Lessons learned from electricity market liberalization. *The Energy Journal*, 29(2):9–42.

[Kaelbling et al., 1998] Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1-2):99–134.

[Kannberg et al., 2003] Kannberg, L. D., Chassin, D. P., Desteese, J. G., and Schienbein, L. A. (2003). GridWise: The Benefits of a Transformed Energy System. Technical Report September, Pacific Northwest Research Laboratory of the US Dept. of Energy.

[Karnouskos and de Holanda, 2009] Karnouskos, S. and de Holanda, T. (2009). Simulation of a smart grid city with software agents. In *Computer Modeling and Simulation*.

[Kearns et al., 1995] Kearns, M., Littman, M. L., and Singh, S. (1995). Graphical Models for Game Theory. *Uncertainty in Artificial Intelligence*.

[Ketter et al., 2010] Ketter, W., Collins, J., and Block, C. (2010). Smart Grid Economics: Policy Guidance through Competitive Simulation. *ERS-2010-043-LIS, Erasmus University*.

[Ketter et al., 2013] Ketter, W., Collins, J., and Reddy, P. (2013). Power TAC: A competitive economic simulation of the smart grid. *Energy Economics*, 39:262–270.

[Ketter et al., 2011] Ketter, W., Collins, J., Reddy, P., and Flath, C. (2011). The Power Trading Agent Competition. Technical Report ERS-2011-011-LIS, RSM Erasmus University, The Netherlands.

[Klusch and Gerber, 2002] Klusch, M. and Gerber, A. (2002). Dynamic coalition formation among rational agents.

[Koller and Friedman, 2009] Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

[Koller and Milch, 2003] Koller, D. and Milch, B. (2003). Multi-Agent Influence Diagrams for Representing and Solving Games. *Games and Economic Behavior*, 45(1):181–221.

[Kolter and Ferreira, 2011] Kolter, J. and Ferreira, J. (2011). A large-scale study on predicting and contextualizing building energy usage. In *AAAI Conf. on Artificial Intelligence (AAAI-11)*.

[Kolter et al., 2010] Kolter, J. Z., Batra, S., and Ng, A. Y. (2010). Energy Disaggregation via Discriminative Sparse Coding. In *Neural Information Processing Systems*.

[Kolter and Ng, 2009] Kolter, J. Z. and Ng, A. Y. (2009). Regularization and feature selection in least-squares temporal difference learning. *Proceedings of the 26th Annual International Conference on Machine Learning ICML 09*, 94305:1–8.

[Konishi and Ray, 2003] Konishi, H. and Ray, D. (2003). Coalition formation as a dynamic process. *Journal of Economic Theory*, 110(1):1–41.

[Liao et al., 2010] Liao, H., Wu, Q., and Jiang, L. (2010). Multi-objective optimization by reinforcement learning for power system dispatch and voltage stability. In *Innovative Smart Grid Technologies Europe*.

[Littlestone and Warmuth, 1994] Littlestone, N. and Warmuth, M. K. (1994). The weighted majority algorithm. *Information and Computation*, 108(2):212–261.

[Littman, 1994] Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the Eleventh International Conference on Machine Learning*.

[Loughran and Kulick, 2004] Loughran, D. and Kulick, J. (2004). Demand-Side Management and Energy Efficiency in the United States. *The Energy Journal, 25 (1)*.

[McKelvey and Palfrey, 1995] McKelvey, R. and Palfrey, T. (1995). Quantal Response Equilibria for Normal Form Games. *Games and Economic Behavior*, 10(1):6–38.

[Melo and Veloso, 2009] Melo, F. S. and Veloso, M. (2009). Learning of Coordination: Exploiting Sparse Interactions in Multiagent Systems. *Autonomous Agents and MultiAgent Systems*, pages 773–780.

[Mitchell, 1997] Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

[Modi et al., 2005] Modi, P., Shen, W., Tambe, M., and Yokoo, M. (2005). ADOPT: Asynchronous Distributed Constraint Optimization with Quality Guarantees. *Artificial Intelligence Journal*.

[Murphy, 2002] Murphy, K. (2002). *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley.

[Murray and Gordon, 2007] Murray, C. and Gordon, G. (2007). Finding Correlated Equilibria in General Sum Stochastic Games. Technical Report June, Carnegie Mellon University.

[NREL, 2012] NREL (2012). Renewable Electricity Futures Study Vol 1. Technical report, National Renewable Energy Laboratory of the US Dept. of Energy.

[Oliehoek et al., 2012] Oliehoek, F. A., Witwicki, S. J., and Kaelbling, L. P. (2012). Influence-Based Abstraction for Multiagent Systems. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12)*.

[Paatero and Lund, 2006] Paatero, J. and Lund, P. (2006). A model for generating household electricity load profiles. *International Journal of Energy Research 30 (5), 273290*.

[Peters et al., 2013] Peters, M., Ketter, W., Saar-Tsechansky, M., and Collins, J. (2013). A reinforcement learning approach to autonomous decision-making in smart electricity markets. *Machine Learning*, 92:539.

[Pindyck and Rubinfeld, 2004] Pindyck, R. and Rubinfeld, D. (2004). *Microeconomics, 6th. Edition*. Pearson Prentice Hall.

[Powell et al., 2011] Powell, W. B., George, A., Berger, J., and Boukhtouta, A. (2011). An Adaptive-learning Framework for Semi-cooperative Multi-agent Coordination. In *IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*.

[Prendergast, 1999] Prendergast, C. (1999). The Provision of Incentives in Firms. *Journal of Economic Literature*, 37(1):7–63.

[Rahimi-Kian et al., 2005] Rahimi-Kian, A., Sadeghi, B., and Thomas, R. (2005). Q-learning based supplier-agents for electricity markets. In *IEEE Power Engineering Society*.

[Ramchurn et al., 2011] Ramchurn, S. D., Vytelingum, P., Rogers, A., and Jennings, N. (2011). Agent-Based Control for Decentralised Demand Side Management in the Smart Grid. In *Autonomous Agents and Multiagent Systems*.

[Ramchurn et al., 2012] Ramchurn, S. D., Vytelingum, P., Rogers, A., and Jennings, N. R. (2012). Putting the Smarts into the Smart Grid: A Grand Challenge for Artificial Intelligence. *ACM Communications*.

[Reddi and Brunskill, 2012] Reddi, S. and Brunskill, E. (2012). Incentive Decision Processes. In *Uncertainty in Artificial Intelligence*.

[Reddy and Veloso, 2011a] Reddy, P. and Veloso, M. (2011a). Learned Behaviors of Multiple Autonomous Agents in Smart Grid Markets. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI-11)*.

[Reddy and Veloso, 2011b] Reddy, P. and Veloso, M. (2011b). RSSI-based Physical Layout Classification and Target Tethering in Mobile Ad-hoc Networks. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-11)*, volume 11.

[Reddy and Veloso, 2011c] Reddy, P. and Veloso, M. (2011c). Strategy Learning for Autonomous Agents in Smart Grid Markets. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI-11)*.

[Reddy and Veloso, 2012] Reddy, P. and Veloso, M. (2012). Factored Models for Multiscale Decision-Making in Smart Grid Customers. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12)*.

[Reddy and Veloso, 2013] Reddy, P. and Veloso, M. (2013). Negotiated Learning for Smart Grid Agents: Entity Selection based on Dynamic Partially Observable Features. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI-13)*.

[Riley, 2005] Riley, P. (2005). *Coaching: Learning and Using Environment and Agent Models for Advice*. PhD thesis, Computer Science Dept., Carnegie Mellon University.

[Ross et al., 2011] Ross, S., Gordon, G., and Bagnell, J. A. (2011). A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. *Artificial Intelligence and Statistics*, 15.

[Roth, 2003] Roth, M. (2003). *Execution-time communication decisions for coordination of multi-agent teams*. PhD thesis, Carnegie Mellon University.

[Roth et al., 2007] Roth, M., Simmons, R., and Veloso, M. (2007). Exploiting factored representations for decentralized execution in multiagent teams. In *Autonomous Agents and Multiagent Systems*.

[Rummery and Niranjan, 1994] Rummery, G. A. and Niranjan, M. (1994). On-line Q-learning using connectionist systems. Technical Report September, Cambridge University Engineering Department.

[Russell and Norvig, 2003] Russell, S. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Prentice Hall.

[San Diego Gas & Electric, 2012] San Diego Gas & Electric (2012). http://www.sdge.com.

[Sandholm and Lesser, 1995] Sandholm, T. and Lesser, V. R. (1995). Coalition Formation among Bounded Rational Agents. *International Joint Conference on Artificial Intelligence*, 14(1):662–669.

[Sandholm and Crites, 1996] Sandholm, T. W. and Crites, R. H. (1996). On multiagent q-learning in a semi-competitive domain. In *Adaption and Learning in Multi-Agent Systems*, pages 191–205. Springer.

[Shapley, 1953] Shapley, L. (1953). Stochastic Games. In *Proceedings of National Academy of Sciences*.

[Shoham and Leyton-Brown, 2009] Shoham, Y. and Leyton-Brown, K. (2009). *Multiagent Systems*. Cambridge University Press.

[Singh et al., 2004] Singh, S., James, M. R., and Rudary, M. R. (2004). Predictive State Representations: A New Theory for Modeling Dynamical Systems. In *Proceedings of the Twentieth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 512–519.

[Skytte, 1999] Skytte, K. (1999). The regulating power market on the Nordic power exchange Nord Pool: an econometric analysis. *Energy Economics*, 21 (4):295308.

[Stahl and Wilson, 1995] Stahl, D. and Wilson, P. (1995). On players' models of other players: Theory and experimental evidence. *Games and Economic Behavior*, 10:218–254.

[Stone and Veloso, 2000] Stone, P. and Veloso, M. (2000). Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8(3):345–383.

[Strbac, 2008] Strbac, G. (2008). Demand side management: benefits and challenges. *Energy Policy 36 (12), 44194426*.

[Strehl and Littman, 2008] Strehl, A. L. and Littman, M. L. (2008). An Analysis of Model-Based Interval Estimation for Markov Decision Processes. *Journal of Computer and System Sciences*, 74(8):1309–1331.

[Sun and Tesfatsion, 2007] Sun, J. and Tesfatsion, L. (2007). An Agent-Based Computational Laboratory for Wholesale Power Market Design. In *IEEE Power and Energy Society General Meeting*.

[Sutton and Barto, 1995] Sutton, R. and Barto, A. (1995). *Reinforcement Learning: An Introduction*. MIT Press.

[Tesfatsion, 2006] Tesfatsion, L. (2006). Agent-Based Computational Economics: A Constructive Approach to Economic Theory. *Handbook of Computational Economics, Vol. 2*.

[United States Department of Energy, 2012] United States Department of Energy (2012). 2010 smart grid system report.

[Voice et al., 2011] Voice, T. D., Vytelingum, P., Ramchurn, S. D., Rogers, A., and Jennings, N. R. (2011). Decentralised Control of Micro-Storage in the Smart Grid. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI-11)*, pages 1421–1427.

[Vytelingum et al., 2010] Vytelingum, P., Ramchurn, S. D., Voice, T. D., Rogers, A., and Jennings, N. R. (2010). Trading Agents for the Smart Electricity Grid. In *Autonomous Agents and Multiagent Systems*.

[Vytelingum et al., 2011] Vytelingum, P., Voice, T., Ramchurn, S., Rogers, A., and Jennings, N. (2011). Theoretical and Practical Foundations of Large-Scale Agent-Based Micro-Storage in the Smart Grid. *Journal of Artificial Intelligence Research*.

[Watkins and Dayan, 1992] Watkins, C. and Dayan, P. (1992). Q-learning. *Machine Learning, 8:279-292*.

[Wellman, 1985] Wellman, M. P. (1985). Reasoning about preference models. Technical report, MIT, MIT/LCS/TR-340.

[Wellman et al., 2007] Wellman, M. P., Greenwald, A., and Stone, P. (2007). *Autonomous Bidding Agents*. MIT Press.

[Wernz and Deshmukh, 2010a] Wernz, C. and Deshmukh, A. (2010a). Multiscale Decision-Making: Bridging Organizational Scales in Systems with Distributed Decision Makers. *Journal of Operational Research 202 828-840*.

[Wernz and Deshmukh, 2010b] Wernz, C. and Deshmukh, A. (2010b). Multiscale decision-making: Bridging organizational scales in systems with distributed decision-makers. *European Journal of Operational Research*, 202(3):828–840.

[West and Harrison, 1997] West, M. and Harrison, P. (1997). *Bayesian Forecasting and Dynamic Models*. Springer-Verlag, 2nd ed.

[Widrow and Hoff, 1960] Widrow, B. and Hoff, M. E. (1960). Adaptive switching circuits. In *1960 IRE WESCON Convention Record*.

[Witwicki and Durfee, 2010] Witwicki, S. J. and Durfee, E. H. (2010). Inuence-based policy abstraction for weakly-coupled Dec-POMDPs. In *Int. Conf. on Automated Planning and Scheduling (ICAPS)*.

[Wright and Leyton-Brown, 2010] Wright, J. R. and Leyton-Brown, K. (2010). Beyond Equilibrium: Predicting Human Behavior in Normal-Form Games. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)*.

[Xiong et al., 2002] Xiong, G., Hashiyama, T., and Okuma, S. (2002). An electricity supplier bidding strategy through Q-Learning. In *IEEE Power Engineering Society*.

MACHINE LEARNING
DEPARTMENT
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213

Carnegie Mellon.